EFFECTS OF FINITE-PRECISION ARITHMETIC ON INTERIOR-POINT METHODS FOR NONLINEAR PROGRAMMING

STEPHEN J. WRIGHT*

Abstract. We show that the effects of finite-precision arithmetic in forming and solving the linear system that arises at each iteration of primal-dual interior-point algorithms for nonlinear programming are benign. When we replace the standard assumption that the active constraint gradients are independent by the weaker Mangasarian-Fromovitz constraint qualification, rapid convergence usually is attainable, even when cancellation and roundoff errors occur during the calculations. In deriving our main results, we prove a key technical result about the size of the exact primal-dual step. This result can be used to modify existing analysis of primal-dual interior-point methods for convex programming, making it possible to extend the superlinear local convergence results to the nonconvex case.

AMS subject classifications. 90C33, 90C30, 49M45

1. Introduction.

We investigate the effects of finite-precision arithmetic on the calculated steps of primal-dual interior-point (PDIP) algorithms for the nonlinear programming problem

$$\text{NLP:} \qquad \min_z \phi(z) \qquad \text{subject to } g(z) \le 0, \tag{1.1}$$

where $\phi : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^n \to \mathbb{R}^m$ are twice Lipschitz continuously differentiable functions. Optimality conditions for this problem can be derived from the Lagrangian function $\mathcal{L}(z, \lambda)$, which is defined as

$$\mathcal{L}(z,\lambda) = \phi(z) + \sum_{i=1}^{m} \lambda_i g_i(z) = \phi(z) + \lambda^T g(z), \tag{1.2}$$

where $\lambda \in \mathbb{R}^m$ is a vector of Lagrange multipliers. When a constraint qualification (discussed below) holds at the point z^*, first-order necessary conditions for z^* to be a solution of (1.1) are that there exists a vector of Lagrange multipliers $\lambda^* \in \mathbb{R}^m$ such that the following conditions are satisfied for $(z, \lambda) = (z^*, \lambda^*)$:

$$\mathcal{L}_z(z,\lambda) = \nabla\phi(z) + \nabla g(z)\lambda = 0, \quad g(z) \le 0, \quad \lambda \ge 0, \quad \lambda^T g(z) = 0. \tag{1.3}$$

These are the well-known Karush-Kuhn-Tucker (KKT) conditions. We use $\mathcal{S}_\lambda$ to denote the set of vectors λ^* such that (z^*, λ^*) satisfies (1.3). The primal-dual solution set is defined by

$$\mathcal{S} = \{z^*\} \times \mathcal{S}_\lambda. \tag{1.4}$$

This paper discusses local convergence of PDIP algorithms for (1.1), assuming that the algorithm is implemented on a computer that performs calculations according to the standard model of floating-point arithmetic. Because of our focus on *local* convergence properties, we assume throughout that the current iterate (z, λ) is close enough to the solution set $\mathcal{S}$ that superlinear convergence would occur if exact steps

---

*Mathematics and Computer Science Division, Argonne National Laboratory, 9700 South Cass Avenue, Argonne, Illinois 60439, U.S.A. This work was supported by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing, U.S. Department of Energy, under Contract W-31-109-Eng-38.

(uncorrupted by finite precision) were taken. In the interests of generality, we weaken an assumption that is often made in the analysis of algorithms for (1.1), namely, that the gradients of the active constraints are linearly independent at the solution. We replace this linear independence constraint qualification (LICQ) with the weaker Mangasarian-Fromovitz constraint qualification (MFCQ) [19]. MFCQ allows constraint gradients to become dependent at the solution, so that the set $\mathcal{S}_\lambda$ of optimal Lagrange multipliers is no longer necessarily a singleton, though it remains bounded. We continue to assume that a strict complementarity (SC) condition holds, that is,

$$g_i(z^*) = 0 \;\Rightarrow\; \lambda_i^* > 0, \qquad \text{for some } \lambda^* \in \mathcal{S}_\lambda. \tag{1.5}$$

In the context of rapidly convergent algorithms, the SC condition makes good sense. If SC fails to hold, superlinear convergence of Newton-like algorithms does not occur, except for specially modified algorithms such as those that identify the active constraints explicitly (see Monteiro and Wright [21] and El-Bakry, Tapia, and Zhang [8]).

The major conclusion of the paper is that the effects of roundoff errors on the rapid local convergence of the algorithm are fairly benign. When a standard second-order condition is added to the assumptions already mentioned, the steps produced under floating-point arithmetic approach $\mathcal{S}$ almost as effectively as do exact steps, as long as the distance to the solution set remains significantly greater than the unit roundoff $\mathbf{u}$. The latter condition is hardly restrictive, since the data errors made in storing the problem in a digital computer mean that the solution set is known only to within some multiple of $\mathbf{u}$ in any case.

The conclusions about the effectiveness of the computed steps are not obvious, because all three formulations of the linear system that must be solved to compute the step at each iteration may become highly ill conditioned near the solution. Our analysis would be significantly simpler if we were to make the LICQ assumption because, in this case, one formulation of the linear equations remains well conditioned, and stability of the three standard formulations can be proved by exploiting their relationship to this system of equations.

This work is related to earlier work of the author on finite-precision analysis of interior-point algorithms for linear complementarity problems [25] and linear programming [28, 31]. The existence of second-order effects gives the analysis here a somewhat different flavor, however. In addition, we go into more depth in checking that the computed iterates can continue to satisfy the approximate centrality conditions usually required in primal-dual algorithms, and in deriving expressions for the rate at which the computed iterates approach the solution set. Related work by Forsgren, Gill, and Shinnerl [9] deals with one formulation of the step equations for the nonlinear programming problem—the so-called augmented form treated here in Section 6—but makes assumptions on the pivot sequence that do not always hold in practice. M. H. Wright [24] recently presented an analysis of the condensed form of the step equations discussed in Section 5 under the assumption that LICQ holds, and found that the computed steps were more accurate than would be expected from a naive analysis.

For linear programming, the PDIP approach has emerged as the most powerful of the interior-point approaches. The supporting theory is strong, in terms of global and local convergence analysis and complexity theory (see the bibliography of Wright [27]). Implementations yield better results than pure-primal or barrier-function approaches; see Andersen et al. [1]. Strong theory is also available for these algorithms when applied to convex programming, in which $\phi(\cdot)$ and $g_i(\cdot)$, $i = 1, \ldots, m$ are all convex functions; see, for example, Wright and Ralph [32] and Ralph and Wright [23, 22]. The latter paper drops the LICQ assumption in favor of MFCQ, making the local theory stronger in one sense than the corresponding local theory for the sequential quadratic programming (SQP) algorithm. The use of MFCQ complicates the analysis considerably, however; under LICQ, the implicit function theorem can be used to prove a key technical result about the length of the step, while more complicated logic is needed to derive this same result under MFCQ.

A significant by-product of the current paper is to prove the key technical result about the length of the rapidly convergent step (Corollary 4.3) under MFCQ and SC, even when the problem (1.1) is not convex. This allows the local convergence results of Ralph and Wright [32, 23, 22] to be extended to general nonconvex nonlinear problems.

The analysis of this paper could also be applied to the recently proposed stabilized sequential quadratic programming (sSQP) algorithm (see Wright [30] and Hager [15]), in which small penalties on the change in the multiplier estimate λ from one iteration to the next ensure rapid convergence even when LICQ is relaxed to MFCQ. A finite-precision analysis of the sSQP method appears in [30, Section 3.2], but only for the augmented form of the step equations. Analysis quite similar to that of the current paper could be applied to show that similar conclusions continue to hold when a condensed form of the step equations is used instead. We omit the details.

The remainder of this paper is structured in the following way. Section 2 contains notation, together with our basic assumptions about (1.1) and some relevant results from the literature. Section 3 discusses the primal-dual interior-point framework, defining the general form of each iteration and the step equations that must be solved at each iteration. Subsection 3.2 proves an important technical result about the relationship between the distance of an interior-point iterate to the solution set $\mathcal{S}$ and a duality measure μ. Section 4 describes perturbed variants of the linear systems that are solved to obtain PDIP steps, and proves our key results about the effect of the perturbations on the accuracy of the steps.

Section 5 focuses on one form of the PDIP step equations: the most compact form in which most of the computational effort goes into factoring a symmetric positive definite matrix, usually by a Cholesky procedure. We trace the effect on step accuracy of errors in evaluation of the functions, formation of the system, and the factorization/solution process. Further, we show the effects of these inaccuracies on the distance that we can move along the steps before the interiority condition is violated, and on various measures of algorithmic progress. An analogous treatment of the augmented form of the step equations appears in Section 6. The conclusions of this section depend on the actual algorithm used to solve the augmented system—it is not sufficient to assume, as in Section 5, that any backward-stable procedure is used to factor the matrix. (We note that similar results hold for the full form of the step equations, but we omit the details of this case, which can be found in the technical report [29].) We conclude with a numerical illustration of the main results in Section 7 and summarize the paper in Section 8.

2. Notation, Assumptions, and Basic Results.

We use $\mathcal{B}$ to denote the set of active indices at z^*, that is,

$$\mathcal{B} = \{i = 1, 2, \ldots, m \,|\, g_i(z^*) = 0\}, \tag{2.1}$$

whereas $\mathcal{N}$ denotes its complement

$$\mathcal{N} = \{1, 2, \ldots, m\} \backslash \mathcal{B}. \tag{2.2}$$

The set $\mathcal{B}_+ \subset \mathcal{B}$ is defined as

$$\mathcal{B}_+ = \{i \in \mathcal{B} \,|\, \lambda_i^* > 0 \text{ for some } \lambda^* \text{ satisfying (1.3)}\}. \tag{2.3}$$

The strict complementarity condition (1.5) is equivalent to

$$\mathcal{B}_+ = \mathcal{B}. \tag{2.4}$$

We frequently make reference to submatrices and subvectors corresponding to the index sets $\mathcal{B}$ and $\mathcal{N}$. For example, the quantities $\lambda_{\mathcal{B}}$ and $g_{\mathcal{B}}(z)$ are the vectors containing the components λ_i and $g_i(z)$, respectively, for $i \in \mathcal{B}$, while $\nabla g_{\mathcal{B}}(z)$ is the Jacobian of $g_{\mathcal{B}}(z)$.

The Mangasarian-Fromovitz constraint qualification (MFCQ) is satisfied at z^*; that is, there is a vector $\bar{y} \in \mathsf{R}^n$ such that

$$\nabla g_{\mathcal{B}}(z^*)^T \bar{y} < 0. \tag{2.5}$$

The following fundamental result about MFCQ is due to Gauvin [11].

LEMMA 2.1. *Suppose that the first-order conditions (1.3) are satisfied at $z = z^*$. Then $\mathcal{S}_\lambda$ is bounded if and only if the MFCQ condition (2.5) is satisfied.*

This result is crucial because it allows our (local) analysis to place a uniform bound on all λ in a neighborhood of the dual solution set $\mathcal{S}_\lambda$.

The second-order condition used in most of the remainder of the paper is that there is a constant $\xi > 0$ such that

$$w^T \mathcal{L}_{zz}(z^*, \lambda^*) w \geq \xi \|w\|^2, \tag{2.6}$$

for all $\lambda^* \in \mathcal{S}_\lambda$ and all w satisfying

$$\begin{aligned} \nabla g_i(z^*)^T w = 0, &\quad \text{for all } i \in \mathcal{B}_+, \\ \nabla g_i(z^*)^T w \leq 0, &\quad \text{for all } i \in \mathcal{B} \backslash \mathcal{B}_+. \end{aligned} \tag{2.7}$$

When the SC condition (1.5) (alternatively, (2.4)) is satisfied, this direction set is simply null $\nabla g_{\mathcal{B}}(z^*)^T$.

A simple example that satisfies MFCQ but not LICQ at the solution, and that satisfies the second-order conditions (2.6), (2.7) and the SC condition is as follows:

$$\min_{z \in \mathsf{R}^2} z_1 \text{ subject to } (z_1 - 1/3)^2 + z_2^2 \leq 1/9, \;\; (z_1 - 2/3)^2 + z_2^2 \leq 4/9. \tag{2.8}$$

The solution is $z^* = 0$, and the optimal multiplier set is

$$\mathcal{S}_\lambda = \{\lambda \geq 0 \,|\, 2\lambda_1 + 4\lambda_2 = 3\}. \tag{2.9}$$

The gradients of the two constraints are the solution are $(-2/3, 0)^T$ and $(-4/3, 0)^T$, respectively. They are linearly independent, but the MFCQ condition (2.5) can be satisfied by choosing $\bar{y} = (1, 0)^T$.

We use $\mathbf{u}$ to denote the unit roundoff, which we define as the smallest number such that the following property holds: When x and y are any two floating-point numbers, op denotes $+$, $-$, $\times$, or $/$, and $\mathit{fl}(z)$ denotes the floating-point approximation of a real number z, we have

$$\mathit{fl}(x \text{ op } y) = (x \text{ op } y)(1 + \epsilon), \qquad |\epsilon| \leq \mathbf{u}. \tag{2.10}$$

Modest multiples of $\mathbf{u}$ are denoted by $\delta_{\mathbf{u}}$.

We assume that the problem is scaled so that the values of g and ϕ and their first and second derivatives in the vicinity of the solution set $\mathcal{S}$, and the values (z, λ) themselves, can all be bounded by moderate quantities. When multiplied by $\mathbf{u}$, quantities of this type are absorbed into the notation $\delta_{\mathbf{u}}$ in the analysis below.

Order notation $O(\cdot)$ and $\Theta(\cdot)$ is used as follows: If v (vector or scalar) and ϵ (nonnegative scalar) are two quantities that share a dependence on other variables, we write $v = O(\epsilon)$ if there is a moderate constant β_1 such that $\|v\| \le \beta_1 \epsilon$ for all values of ϵ that are interesting in the given context. (The "interesting context" frequently includes cases in which ϵ is either sufficiently small or sufficiently large, but we often use $v = O(\mu)$ to indicate that $\|v\| \le \beta_1 \mu$ for all sufficiently small μ that satisfy $\mu \gg \mathbf{u}$, for some β_1; see later discussion.) We write $v = \Theta(\epsilon)$ if there are constants β_1 and β_0 such that $\beta_0 \epsilon \le \|v\| \le \beta_1 \epsilon$ for all interesting values of ϵ. Similarly, we write $v = O(1)$ if $\|v\| \le \beta_1$, and $v = \Theta(1)$ if $\beta_0 \le \|v\| \le \beta_1$.

We use the notation $\delta(z, \lambda)$ to denote the distance from (z, λ) to the primal-dual solution set, that is,

$$\delta(z,\lambda) \overset{\text{def}}{=} \min_{(z^*,\lambda^*)\in\mathcal{S}} \|(z,\lambda) - (z^*,\lambda^*)\|. \tag{2.11}$$

It is well known (see, for example, Theorem A.1 of Wright [26]) that this distance can be estimated in terms of known quantities at (z, λ). We state this result formally as follows.

THEOREM 2.2. *Suppose that the MFCQ condition (2.5) and the second-order conditions (2.6), (2.7) are satisfied at the solution z^*. Then if $\lambda \ge 0$, we have*

$$\delta(z,\lambda) = \Theta\left(\left\| \begin{bmatrix} \mathcal{L}_z(z,\lambda) \\ \min(\lambda, -g(z)) \end{bmatrix} \right\| \right). \tag{2.12}$$

We write the singular value decomposition (SVD) of the Jacobian of the active constraints as follows:

$$\nabla g_{\mathcal{B}}(z^*) = \begin{bmatrix} \hat{U} & \hat{V} \end{bmatrix} \begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U^T \\ V^T \end{bmatrix} = \hat{U} \Sigma U^T, \tag{2.13}$$

where the matrices $\begin{bmatrix} \hat{U} & \hat{V} \end{bmatrix}$ and $\begin{bmatrix} U & V \end{bmatrix}$ are orthogonal, and Σ is a diagonal matrix with positive diagonal elements.

Note that the columns of $\hat{U}$ form a basis for the range space of $\nabla g_{\mathcal{B}}(z^*)$, while the columns of $\hat{V}$ form a basis for the null space of $\nabla g_{\mathcal{B}}(z^*)^T$. Similarly, the columns of U form a basis for the range space of $\nabla g_{\mathcal{B}}(z^*)^T$, while the columns of V form a basis for the null space of $\nabla g_{\mathcal{B}}(z^*)$. These four subspaces are key to our analysis, particularly the subspace spanned by the columns of V. For the computational methods used to solve the primal-dual step equations discussed in this paper, the computed step in the $\mathcal{B}$-components of the multipliers—that is, $\Delta\lambda_{\mathcal{B}}$—has a larger error in the range space of V than in the complementary subspace spanned by the columns of U. The errors in the computed primal step Δz, the computed step of the $\mathcal{N}$-components of the multipliers $\lambda_{\mathcal{N}}$, and the computed step in the dual slack variables (defined later) are typically also less significant than the error in $V^T \Delta\lambda_{\mathcal{B}}$. We show, however, that the potentially large error in $V^T \Delta\lambda_{\mathcal{B}}$ does not affect the performance of primal-dual algorithms that use these computed steps until μ becomes similar to $\mathbf{u}^{1/2}$.

When the stronger LICQ condition holds, the matrices $\hat{V}$ and V are vacuous, and the SVD (2.13) reduces to $\nabla g_{\mathcal{B}}(z^*) = \hat{U}\Sigma U^T$. Much of the analysis in this paper would simplify considerably under LICQ, in part because $V^T \Delta\lambda_{\mathcal{B}}$—the step component with the large error—is no longer present.

We use $\sigma_{\min}(\cdot)$ to denote the smallest eigenvalue, and $\text{cond}(\cdot)$ to denote the condition number, as measured by the Euclidean norm.

3. Primal-Dual Interior-Point Methods.

3.1. Centrality Conditions and Step Equations. Primal-dual interior-point methods are constrained, modified Newton methods applied to a particular form of the KKT conditions (1.3). By introducing a vector $s \in \mathbb{R}^m$ of slacks for the inequality constraint, we can rewrite the nonlinear program as

$$\min_{(z,s)} \phi(z) \quad \text{subject to } g(z) + s = 0,\ s \geq 0,$$

and the KKT conditions (1.3) as

$$\mathcal{L}_z(z,\lambda) = 0, \quad g(z) + s = 0, \quad (\lambda, s) \geq 0, \quad \lambda^T s = 0. \tag{3.1}$$

Motivated by this form of the conditions, we define the mapping $\mathcal{F}(z,\lambda,s)$ by

$$\mathcal{F}(z,\lambda,s) \overset{\text{def}}{=} \begin{bmatrix} \nabla\phi(z) + \nabla g(z)\lambda \\ g(z) + s \\ S\Lambda e \end{bmatrix}, \tag{3.2}$$

where the diagonal matrices S and Λ are defined by

$$S \overset{\text{def}}{=} \text{diag}(s_1, s_2, \ldots, s_m), \qquad \Lambda \overset{\text{def}}{=} \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m),$$

and e is defined as

$$e = (1, 1, \ldots, 1)^T. \tag{3.3}$$

The KKT conditions (3.1) can now be stated equivalently as

$$\mathcal{F}(z,\lambda,s) = 0, \qquad (s,\lambda) \geq 0. \tag{3.4}$$

Primal-dual iterates (z,λ,s) invariably satisfy the strict bound $(s,\lambda) > 0$, while they approach satisfaction of the condition $\mathcal{F}(\cdot) = 0$ in the limit. An important measure of progress is the *duality measure* $\mu(\lambda, s)$, which is defined by

$$\mu(\lambda,s) \overset{\text{def}}{=} \lambda^T s/m. \tag{3.5}$$

When μ is used without arguments, we assume that $\mu = \mu(\lambda, s)$, where (z,λ,s) is the current primal-dual iterate. We emphasize that μ is a function of (z,λ,s), rather than a target value explicitly chosen by the algorithm, as is the case in some of the literature.

A typical step $(\Delta z, \Delta\lambda, \Delta s)$ of the primal-dual method satisfies

$$\nabla\mathcal{F}(z,\lambda,s) \begin{bmatrix} \Delta z \\ \Delta\lambda \\ \Delta s \end{bmatrix} = -\mathcal{F}(z,\lambda,s) - \begin{bmatrix} 0 \\ 0 \\ t \end{bmatrix}, \tag{3.6}$$

where t defines the deviation from a pure Newton step for $\mathcal{F}$ (which is also known as a "primal-dual affine-scaling" step). The vector t frequently contains a centering term $\sigma\mu e$, where σ is a centering parameter in the range $[0,1]$. It sometimes also contains higher-order information, such as the product $\Delta\Lambda_{\mathrm{aff}}\Delta S_{\mathrm{aff}}e$, where $\Delta\Lambda_{\mathrm{aff}}$ and ΔS_{aff} are the diagonal matrices constructed from the components of the pure Newton step (Mehrotra [20]). In any case, the vector t usually goes to zero rapidly as the iterates converge to a solution, so that the steps generated from (3.6) approach pure Newton steps, which in turn ensures rapid local convergence. Throughout this paper, we assume that t satisfies the estimate

$$t = O(\mu^2). \tag{3.7}$$

Using the definition (1.2), we can write the system (3.6) explicitly as follows:

$$\begin{bmatrix} \mathcal{L}_{zz}(z,\lambda) & \nabla g(z) & 0 \\ \nabla g(z)^T & 0 & I \\ 0 & S & \Lambda \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta\lambda \\ \Delta s \end{bmatrix} = - \begin{bmatrix} \mathcal{L}_z(z,\lambda) \\ g(z)+s \\ S\Lambda e + t \end{bmatrix}. \tag{3.8}$$

Block eliminations can be performed on this system to yield more compact formulations. By eliminating Δs, we obtain the *augmented system* form, which is

$$\begin{bmatrix} \mathcal{L}_{zz}(z,\lambda) & \nabla g(z) \\ \nabla g(z)^T & -\Lambda^{-1}S \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta\lambda \end{bmatrix} = \begin{bmatrix} -\mathcal{L}_z(z,\lambda) \\ -g(z)+\Lambda^{-1}t \end{bmatrix}. \tag{3.9}$$

By eliminating $\Delta\lambda$ from this system, we obtain a system that is sometimes referred to as the *condensed* form (or in the case of linear programming as the *normal equations* form), which is

$$\begin{aligned} &\left[\mathcal{L}_{zz}(z,\lambda) + \nabla g(z)\Lambda S^{-1}\nabla g(z)^T\right]\Delta z \\ &\quad = -\mathcal{L}_z(z,\lambda) - \nabla g(z)\Lambda S^{-1}[g(z) - \Lambda^{-1}t]. \end{aligned} \tag{3.10}$$

We consider primal-dual methods in which each iterate (z,λ,s) satisfies the following properties:

$$\|r_f(z,\lambda)\| \le C\mu, \quad \text{where } r_f(z,\lambda) \overset{\text{def}}{=} \mathcal{L}_z(z,\lambda), \tag{3.11a}$$

$$\|r_g(z,s)\| \le C\mu, \quad \text{where } r_g(z,s) \overset{\text{def}}{=} g(z)+s, \tag{3.11b}$$

$$(\lambda,s) > 0, \quad \lambda_i s_i \ge \gamma\mu, \quad \text{for all } i = 1,2,\ldots,m, \tag{3.11c}$$

for some constants $C > 0$ and $\gamma \in (0,1)$, where μ is defined as in (3.5). (In much of the succeeding discussion, we omit the arguments from the quantities from μ, r_f, and r_g when they are evaluated at the current iterate (z,λ,s).) These conditions ensure that the pairwise products $s_i\lambda_i$, $i = 1,2,\ldots,m$ are not too disparate and that the first two components of $\mathcal{F}$ in (3.2) can be bounded in terms of the third component. They are sometimes called the *centrality conditions* because they are motivated by the notion of a central path and its neighborhoods. Conditions of the type (3.11) are imposed in most path-following interior-point methods for linear programming (see, for example, [27]). For nonlinear convex programming, examples of methods that require these conditions can be found in Ralph and Wright [32, 23, 22]. In nonlinear programming, we mention Gould et al. [14] (see Algorithm 4.1 and Figure 5.1) and Byrd, Liu, and Nocedal [4]. In the latter paper, (3.11a) and (3.11b) are imposed explicitly, while (3.11c) can be guaranteed by choosing $\epsilon_\mu = (1-\gamma)\mu$. Even when the choice $\epsilon_\mu = \mu$ is made, as in the bulk of the discussion in [4], their other conditions concerning positivity of (s, λ) can be expected to produce iterates that satisfy (3.11c) in practice.

For points (z, λ, s) that satisfy (3.11), we can use μ to estimate the distance $\delta(z, \lambda)$ from (z, λ) to the solution set $\mathcal{S}$ (see (2.11)). These results, which are proved in the following subsection, can be summarized briefly as follows. When the MFCQ condition (2.5) and the second-order conditions (2.6), (2.7) are satisfied, we have that $\delta(z, \lambda) = O(\mu^{1/2})$. When the strict complementarity assumption (1.5) is added, we obtain the stronger estimate $\delta(z, \lambda) = O(\mu)$. We can use these estimates to obtain bounds on the elements of the diagonal matrices S, Λ, $S^{-1}\Lambda$, and $\Lambda^{-1}S$ in the systems above; these bounds are the key to the error analysis of the remainder of the paper.

3.2. Using the Duality Measure to Estimate Distance to the Solution.

The main result of this section, Theorem 3.3, shows that under certain assumptions, the distance $\delta(z, \lambda)$ of a primal-dual iterate (z, λ, s) to the solution set $\mathcal{S}$ can be estimated by the duality measure μ. We start with a technical lemma that proves the weaker estimate $\delta(z, \lambda) = O(\mu^{1/2})$. Note that this result does not assume that the SC condition (1.5) holds.

Lemma 3.1. *Suppose that z^* is a solution of (1.1) at which the MFCQ condition (2.5) and the second-order conditions (2.6), (2.7) are satisfied. Then for all (z, λ) with $\lambda \geq 0$ for which there is a vector s such that (z, λ, s) satisfies (3.11), we have that*

$$\delta(z, \lambda) = O(\mu^{1/2}). \tag{3.12}$$

Proof. We prove the result by showing that

$$\begin{bmatrix} \mathcal{L}_z(z, \lambda) \\ \min(\lambda, -g(z)) \end{bmatrix} = O(\mu^{1/2}) \tag{3.13}$$

and then applying Theorem 2.2. Since $\mathcal{L}_z(z, \lambda) = r_f = O(\mu)$, the first part of the vector satisfies the required estimate. For the second part, we have from (3.11b) that

$$-g(z) = s - r_g = s + O(\mu),$$

and hence that

$$\min(-g_i(z), \lambda_i) = \min(s_i, \lambda_i) + O(\mu). \tag{3.14}$$

Because of (3.5) and (3.11c), we have that $s_i \lambda_i \leq m\mu$ and $(\lambda_i, s_i) > 0$. It follows immediately that $\min(\lambda_i, s_i) \leq (m\mu)^{1/2}$ for $i = 1, 2, \ldots, m$. Hence, by substitution into (3.14), we obtain

$$\min(-g_i(z), \lambda_i) \leq (m\mu)^{1/2} + O(\mu) = O(\mu^{1/2}).$$

We conclude that the second part of the vector in (3.13) is of size $O(\mu^{1/2})$, so the proof is complete. $\square$

The following examples show the upper bound of Lemma 3.1 is indeed achieved and that it is not possible to obtain a lower bound on $\delta(z, \lambda)$ as a function of μ. To demonstrate the first claim, consider the problem

$$\min \tfrac{1}{2} z^2 \qquad \text{subject to } -z \leq 0.$$

The point $(z,\lambda,s)=(\epsilon,\epsilon,\epsilon)$ satisfies

$$\mathcal{L}_z(z,\lambda)=0,\quad g(z)+s=0,\quad s\lambda=\epsilon^2,\quad \mu=\epsilon^2,$$

so that the conditions (3.11) are satisfied. Clearly the distance from the point (z,λ) to the solution set $\mathcal{S}=(0,0)$ is $\sqrt{2}\epsilon=\sqrt{2}\mu^{1/2}$. For the second claim, consider any nonlinear program such that $\mathcal{B}=\{1,2,\ldots,m\}$ (that is, all constraints active) and strict complementarity (1.5) holds at some multiplier λ^*. Then for appropriate choices of γ and C, the point

$$(z,\lambda,s)=(z^*,\lambda^*,(m\mu)/(e^T\lambda^*)e) \tag{3.15}$$

satisfies the definition (3.5) and the condition (3.11) for any $\mu>0$. On the other hand, $\delta(z,\lambda)=0$ in (3.15) independently of s and μ.

We now prove an extension of Lemma 5.1 of Ralph and Wright [23], dropping the monotonicity assumption of this earlier result.

LEMMA 3.2. *Suppose that the conditions of Lemma 3.1 hold and in addition that the SC condition (1.5) is satisfied. Then for all (z,λ,s) satisfying (3.11), we have that*

$$i\in\mathcal{B}\Rightarrow s_i=\Theta(\mu),\ \ \lambda_i=\Theta(1), \tag{3.16a}$$

$$i\in\mathcal{N}\Rightarrow s_i=\Theta(1),\ \ \lambda_i=\Theta(\mu). \tag{3.16b}$$

Proof. By boundedness of $\mathcal{S}$ (Lemma 2.1), we have for all (z,λ,s) sufficiently close to $\mathcal{S}$ that

$$\lambda_i=O(1),\qquad s_i=-g_i(z)+(r_g)_i=O(1). \tag{3.17}$$

Given (z,λ,s) satisfying (3.11), let $P(\lambda)$ be the projection of λ onto the set $\mathcal{S}_\lambda$, and let $\lambda^*\in\mathcal{S}_\lambda$ be some strictly complementary optimal multiplier (for which (1.5) is satisfied). From Lemma 3.1 we obtain

$$\|z-z^*\|=O(\mu^{1/2}). \tag{3.18}$$

Using this observation together with smoothness of $\phi(\cdot)$ and $g(\cdot)$, we have for the gradient of $\mathcal{L}$ that

$$\begin{aligned}&\mathcal{L}_z(z,\lambda)-\mathcal{L}_z(z^*,\lambda^*)\\&=\nabla\phi(z)-\nabla\phi(z^*)+\nabla g(z)\lambda-\nabla g(z^*)\lambda^*\\&=O(\mu^{1/2})+\nabla g(z)[\lambda-P(\lambda)]+[\nabla g(z)-\nabla g(z^*)]P(\lambda)+\nabla g(z^*)[P(\lambda)-\lambda^*].\end{aligned}$$

Since $P(\lambda)$ and λ^* are both in $\mathcal{S}_\lambda$, we find from (1.3) that the last term vanishes. From (3.18) and $P(\lambda)=O(1)$, the second-to-last term has size $O(\mu^{1/2})$. For the remaining term, we have $\nabla g(z)=O(1)$, and $\|\lambda-P(\lambda)\|\le\delta(z,\lambda)=O(\mu^{1/2})$. By assembling all these observations, and using $\mathcal{L}_z(z^*,\lambda^*)=0$, we obtain

$$\mathcal{L}_z(z,\lambda)=\mathcal{L}_z(z,\lambda)-\mathcal{L}_z(z^*,\lambda^*)=O(\mu^{1/2}). \tag{3.19}$$

Using again that $\nabla g(z^*)[P(\lambda)-\lambda^*]=0$, we have from (3.18) that

$$\begin{aligned}[P(\lambda)-\lambda^*]^T[g(z)-g(z^*)]&=[P(\lambda)-\lambda^*]^T[\nabla g(z^*)^T(z-z^*)+O(\|z-z^*\|^2)]\\&=O(\|z-z^*\|^2)=O(\mu).\end{aligned} \tag{3.20}$$

By gathering the estimates (3.12), (3.18), (3.19), and (3.20), we obtain

$$\begin{aligned}
&\begin{bmatrix} z-z^* \\ \lambda-\lambda^* \end{bmatrix}^T \begin{bmatrix} \mathcal{L}_z(z,\lambda)-\mathcal{L}_z(z^*,\lambda^*) \\ -g(z)+g(z^*) \end{bmatrix} \\
&= \begin{bmatrix} z-z^* \\ \lambda-P(\lambda) \end{bmatrix}^T \begin{bmatrix} \mathcal{L}_z(z,\lambda)-\mathcal{L}_z(z^*,\lambda^*) \\ -g(z)+g(z^*) \end{bmatrix} \\
&\quad +[P(\lambda)-\lambda^*]^T[-g(z)+g(z^*)] \\
&= O(\delta(z,\lambda))O(\mu^{1/2})+O(\mu)=O(\mu). \qquad (3.21)
\end{aligned}$$

By substituting from (3.11) and using (3.21), we obtain

$$\begin{bmatrix} z-z^* \\ \lambda-\lambda^* \end{bmatrix}^T \begin{bmatrix} r_f \\ s-r_g-s^* \end{bmatrix} = \begin{bmatrix} z-z^* \\ \lambda-\lambda^* \end{bmatrix}^T \begin{bmatrix} \mathcal{L}_z(z,\lambda)-\mathcal{L}_z(z^*,\lambda^*) \\ -g(z)+g(z^*) \end{bmatrix} = O(\mu),$$

and therefore

$$(\lambda-\lambda^*)^T(s-s^*) = -(z-z^*)^T r_f + (\lambda-\lambda^*)^T r_g + O(\mu).$$

By using the conditions (3.11a), (3.11b), and the definition (3.5), we obtain

$$\begin{aligned}
&-\sum_{i=1}^m \lambda_i^* s_i - \sum_{i=1}^m \lambda_i s_i^* \\
&= -(\lambda^*)^T s - \lambda^T s^* = -\lambda^T s + O(\mu) + O(\|z-z^*\|\|r_f\|) + O(\|\lambda-\lambda^*\|\|r_g\|) = O(\mu).
\end{aligned}$$

Since $(\lambda, s) > 0$ and $(\lambda^*, s^*) \ge 0$, all terms $\lambda_i^* s_i$ and $\lambda_i s_i^*$, $i = 1, 2, \ldots, m$ are nonnegative, so there is a constant $C_1 > 0$ such that

$$0 \le \lambda_i^* s_i \le C_1\mu, \quad 0 \le \lambda_i s_i^* \le C_1\mu, \quad \text{for all } i = 1, 2, \ldots, m.$$

For all $i \in \mathcal{B}$, we have $\lambda_i^* > 0$ by our strictly complementary choice of λ^*, and so

$$0 < s_i \le \frac{C_1}{\lambda_i^*}\mu \le \frac{C_1}{\min_{i\in\mathcal{B}} \lambda_i^*}\mu \overset{\text{def}}{=} C_2\mu. \qquad (3.22)$$

On the other hand, we have by boundedness of $\mathcal{S}_\lambda$ and our assumption (3.11c) that

$$s_i \ge \frac{\gamma\mu}{\lambda_i} \ge \gamma_{\min}\mu, \quad \text{for all } i = 1, 2, \ldots, m, \qquad (3.23)$$

for some constant $\gamma_{\min} > 0$. By combining (3.22) and (3.23), we have that

$$s_i = \Theta(\mu), \quad \text{for all } i \in \mathcal{B}.$$

For the $\lambda_{\mathcal{B}}$ component, we have that

$$s_i\lambda_i \ge \gamma\mu \;\Rightarrow\; \lambda_i \ge \frac{\gamma\mu}{s_i} \ge \frac{\gamma}{C_2}, \quad \text{for all } i \in \mathcal{B}.$$

Hence, by combining with (3.17), we obtain that

$$\lambda_i = \Theta(1), \quad \text{for all } i \in \mathcal{B}.$$

This completes the proof of (3.16a). We omit the proof of (3.16b), which is similar. □

Next, we show that when the strict complementarity assumption is added to the assumptions of Lemma 3.1, the upper bound on the distance to the solution set in (3.12) can actually be improved to $O(\mu)$.

THEOREM 3.3. *Suppose that* z^* *is a solution of (1.1) at which the MFCQ condition (2.5), the second-order conditions (2.6), (2.7), and the SC condition (1.5) are satisfied. Then for all* (z, λ) *with* $\lambda \geq 0$ *for which there is a vector* s *such that* (z, λ, s) *satisfies (3.11), we have that*

$$\delta(z, \lambda) = O(\mu). \tag{3.24}$$

Proof. From (3.11a), we have directly that $r_f = O(\mu)$. We have from (3.11) and (3.16a) that

$$g_i(z) = -s_i + (r_g)_i = O(\mu), \quad \lambda_i = \Theta(1), \quad \lambda_i > 0 \quad \text{for all } i \in \mathcal{B},$$

so that

$$\min(-g_i(z), \lambda_i) = -g_i(z) = O(\mu), \quad \text{for all } i \in \mathcal{B}, \tag{3.25}$$

whenever μ is sufficiently small. For the remaining components, we have

$$\min(-g_i(z), \lambda_i) = \lambda_i = O(\mu), \quad \text{for all } i \in \mathcal{N}. \tag{3.26}$$

By substituting (3.11a), (3.25), and (3.26) into (2.12), we obtain the result. □

Similar conclusions to Lemma 3.2 and Theorem 3.3 can be reached in the case of linear programming algorithms. The second-order conditions (2.6), (2.7) are not relevant for this class of problems, and the SC assumption (1.5) holds for every linear program that has a solution.

4. Accuracy of PDIP Steps: General Results.

By partitioning the constraint index set $\{1, 2, \ldots, m\}$ into active indices $\mathcal{B}$ and inactive indices $\mathcal{N}$, we can express the system (3.9) without loss of generality as follows:

$$\begin{bmatrix} \mathcal{L}_{zz}(z,\lambda) & \nabla g_{\mathcal{B}}(z) & \nabla g_{\mathcal{N}}(z) \\ \nabla g_{\mathcal{B}}(z)^T & -D_{\mathcal{B}} & 0 \\ \nabla g_{\mathcal{N}}(z)^T & 0 & -D_{\mathcal{N}} \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \lambda_{\mathcal{B}} \\ \Delta \lambda_{\mathcal{N}} \end{bmatrix} = \begin{bmatrix} -\mathcal{L}_z(z,\lambda) \\ -g_{\mathcal{B}}(z) + \Lambda_{\mathcal{B}}^{-1} t_{\mathcal{B}} \\ -g_{\mathcal{N}}(z) + \Lambda_{\mathcal{N}}^{-1} t_{\mathcal{N}} \end{bmatrix}, \tag{4.1}$$

where $D_{\mathcal{B}}$ and $D_{\mathcal{N}}$ are positive diagonal matrices defined by

$$D_{\mathcal{B}} = \Lambda_{\mathcal{B}}^{-1} S_{\mathcal{B}}, \qquad D_{\mathcal{N}} = \Lambda_{\mathcal{N}}^{-1} S_{\mathcal{N}}. \tag{4.2}$$

When the SC condition (1.5) is satisfied, we have from (3.16) that the diagonals of $D_{\mathcal{B}}$ have size $\Theta(\mu)$ while those of $D_{\mathcal{N}}$ have size $\Theta(\mu^{-1})$. By eliminating $\Delta\lambda_{\mathcal{N}}$ from (4.1), we obtain the following intermediate stage between (3.9) and (3.10):

$$\begin{aligned} &\begin{bmatrix} H(z,\lambda) & \nabla g_{\mathcal{B}}(z) \\ \nabla g_{\mathcal{B}}(z)^T & -D_{\mathcal{B}} \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \lambda_{\mathcal{B}} \end{bmatrix} \\ &= \begin{bmatrix} -\mathcal{L}_z(z,\lambda) - \nabla g_{\mathcal{N}}(z) D_{\mathcal{N}}^{-1} [g_{\mathcal{N}}(z) - \Lambda_{\mathcal{N}}^{-1} t_{\mathcal{N}}] \\ -g_{\mathcal{B}}(z) + \Lambda_{\mathcal{B}}^{-1} t_{\mathcal{B}} \end{bmatrix}, \end{aligned} \tag{4.3}$$

where we have defined

$$H(z,\lambda) \overset{\text{def}}{=} \mathcal{L}_{zz}(z,\lambda) + \nabla g_{\mathcal{N}}(z) D_{\mathcal{N}}^{-1} \nabla g_{\mathcal{N}}(z)^T. \tag{4.4}$$

In this section, we start by proving a key result about the solutions of perturbed forms of the system (4.3). Subsequently, we use this result as the foundation for proving results about the three alternative formulations (3.8), (3.9), and (3.10) of the PDIP step equations. The principal reason for our focus on (4.3) is that the proof of the main result can be derived from fairly standard linear algebra arguments. Gould [13, Section 6] obtains a system similar to (4.3) for the Newton equations for the primal log-barrier function, and notes that the matrix approaches a nonsingular limit when certain optimality conditions, including LICQ, are satisfied. Because we replace LICQ by MFCQ, the matrix in (4.3) may approach a singular limit in our case.

We note that the form (4.3) is also relevant to the stabilized sequential quadratic programming (sSQP) method of Wright [30] and Hager [15]; that is, slight modifications to the results of this paper can be used to show that the condensed and augmented formulations of the step equations for the sSQP algorithm yield good steps even in the presence of roundoff errors and cancellation. We omit further details in this paper.

Errors in the step equations arise from cancellation and roundoff errors in evaluating both the matrix and right-hand side and from roundoff errors that arise in the factorization/solution process. We discuss these sources of error further and quantify them in the next section. In this section, we consider the following perturbed version of (4.3):

$$\begin{bmatrix} H(z,\lambda)+\tilde{E}_{11} & \nabla g_{\mathcal{B}}(z)+\tilde{E}_{12} \\ \nabla g_{\mathcal{B}}(z)^T+\tilde{E}_{21} & -D_{\mathcal{B}}+\tilde{E}_{22} \end{bmatrix} \begin{bmatrix} w \\ y \end{bmatrix} = \begin{bmatrix} r_1 \\ \nabla g_{\mathcal{B}}(z^*)^T r_3 + r_4 \end{bmatrix}. \tag{4.5}$$

Here, $\tilde{E}$ is the perturbation matrix (appropriately partitioned and not assumed to be symmetric) and r_1, r_3, and r_4 represent components of a general right-hand side. Note the partitioning of the second right-hand side component into a component $\nabla g_{\mathcal{B}}(z^*)^T r_3$ in the range space of $\nabla g_{\mathcal{B}}(z^*)^T$ and a remainder term r_4. When LICQ is satisfied, the range space of $\nabla g_{\mathcal{B}}(z^*)^T$ spans the full space, so we can choose r_4 to be zero. Under MFCQ, however, we have in general that r_4 must be nonzero. The main interest of the results below is in isolating the component of the solution of (4.5) that is sensitive to r_4.

To make the results applicable to a wider class of linear systems, we do not impose the assumptions that were needed in the preceding section to ensure that the matrices $D_{\mathcal{B}}$ and $D_{\mathcal{N}}$ defined by (4.2) have diagonals of the appropriate size. Instead, we *assume* that the diagonals have the given size, and derive the application to the linear systems of interest (those that arise in primal-dual interior-point methods) as a special case.

Our results in this and later sections make extensive use of the SVD (2.13) of $\nabla g_{\mathcal{B}}(z^*)$. They also make assumptions about the size of the smallest eigenvalue of the matrix $\hat{V}^T \mathcal{L}_{zz}(z^*,\lambda^*)\hat{V}$, the two-sided projection of the Lagrangian Hessian onto the active constraint manifold.

THEOREM 4.1. *Let (z,λ) be an approximate primal-dual solution of (1.1) with $\delta(z,\lambda)=O(\mu)$, and suppose the diagonal matrices $D_{\mathcal{B}}$ and $D_{\mathcal{N}}^{-1}$ defined by (4.2) have all their diagonal elements of size $\Theta(\mu)$. Then if the perturbation submatrices in (4.5) satisfy*

$$\tilde{E}_{11} = \delta_{\mathbf{u}}/\mu + O(\mu), \qquad \tilde{E}_{21}, \tilde{E}_{12}, \tilde{E}_{22} = \delta_{\mathbf{u}}, \tag{4.6}$$

there is a constant $\beta > 0$ such that when the following conditions hold:

$$\mathbf{u}/\mu \ll 1, \quad \mathbf{u} \ll 1, \tag{4.7a}$$

$$\sigma_{\min}(\Sigma) \gg \beta \max(\mu^{1/3}, \mathbf{u}/\mu), \tag{4.7b}$$

$$\sigma_{\min}(\hat{V}^T \mathcal{L}_{zz}(z^*, \lambda^*)\hat{V}) \gg \beta \max(\mu^{1/3}, \mathbf{u}/\mu), \quad \text{for all } \lambda^* \in \mathcal{S}_\lambda. \tag{4.7c}$$

the step (w, y) computed from (4.5) satisfies

$$\begin{aligned}(U^T y, \hat{V}^T w, \hat{U}^T w) &= O(\|r_1\| + \|r_3\| + \|r_4\|), \\ V^T y &= O(\|r_1\| + \|r_3\| + \|r_4\|/\mu).\end{aligned}$$

Proof. If we define

$$y_U = U^T y, \quad y_V = V^T y, \quad w_{\hat{U}} = \hat{U}^T w, \quad w_{\hat{V}} = \hat{V}^T w,$$

we have

$$y = U y_U + V y_V, \qquad w = \hat{U} w_{\hat{U}} + \hat{V} w_{\hat{V}}.$$

Using this notation, we can rewrite (4.5) as

$$\begin{bmatrix} \hat{U}^T M_{11} \hat{U} & \hat{U}^T M_{11} \hat{V} & \hat{U}^T M_{12} U & \hat{U}^T M_{12} V \\ \hat{V}^T M_{11} \hat{U} & \hat{V}^T M_{11} \hat{V} & \hat{V}^T M_{12} U & \hat{V}^T M_{12} V \\ U^T M_{21} \hat{U} & U^T M_{21} \hat{V} & U^T M_{22} U & U^T M_{22} V \\ V^T M_{21} \hat{U} & V^T M_{21} \hat{V} & V^T M_{22} U & V^T M_{22} V \end{bmatrix} \begin{bmatrix} w_{\hat{U}} \\ w_{\hat{V}} \\ y_U \\ y_V \end{bmatrix} = \begin{bmatrix} \hat{U}^T r_1 \\ \hat{V}^T r_1 \\ U^T \nabla g_{\mathcal{B}}(z^*)^T r_3 + U^T r_4 \\ V^T \nabla g_{\mathcal{B}}(z^*)^T r_3 + V^T r_4 \end{bmatrix}, \tag{4.8}$$

where we have defined

$$\begin{aligned} M_{11} &= H(z, \lambda) + \tilde{E}_{11}, \quad M_{12} = \nabla g_{\mathcal{B}}(z) + \tilde{E}_{12}, \\ M_{21} &= \nabla g_{\mathcal{B}}(z)^T + \tilde{E}_{21}, \quad M_{22} = -D_{\mathcal{B}} + \tilde{E}_{22}, \end{aligned} \tag{4.9}$$

and $H(\cdot, \cdot)$ is defined in (4.4). From (2.13), we have

$$V^T \nabla g_{\mathcal{B}}(z^*)^T = 0, \quad U^T \nabla g_{\mathcal{B}}(z^*)^T = \Sigma \hat{U}^T.$$

The fact that V^T annihilates $\nabla g_{\mathcal{B}}(z^*)^T$ is crucial, because it causes the term with r_3 to disappear from the last component of the right-hand side of (4.8), which becomes

$$\begin{bmatrix} \hat{U}^T r_1 \\ \hat{V}^T r_1 \\ \Sigma \hat{U}^T r_3 + U^T r_4 \\ V^T r_4 \end{bmatrix}. \tag{4.10}$$

From the definitions (4.9) and (4.4), the perturbation bound (4.6), our assumptions that $D_{\mathcal{N}}^{-1} = O(\mu)$ and $\delta(z, \lambda) = O(\mu)$, compactness of $\mathcal{S}$, and the fact that $\mathcal{L}_{zz}$ is Lipschitz continuous, we have that

$$M_{11} = \mathcal{L}_{zz}(z^*, \lambda^*) + \delta_{\mathbf{u}}/\mu + O(\mu), \tag{4.11}$$

for some $\lambda^* \in \mathcal{S}_\lambda$. Using these same facts, we have likewise that

$$M_{21} = \nabla g_{\mathcal{B}}(z^*)^T + \delta_{\mathbf{u}} + O(\mu),$$

so by substituting from (2.13), we have that

$$U^T M_{21} \hat{U} = \Sigma + \delta_{\mathbf{u}} + O(\mu), \quad U^T M_{21} \hat{V} = \delta_{\mathbf{u}} + O(\mu), \tag{4.12a}$$

$$V^T M_{21} \hat{U} = \delta_{\mathbf{u}} + O(\mu), \quad V^T M_{21} \hat{V} = \delta_{\mathbf{u}} + O(\mu). \tag{4.12b}$$

Similarly, from the definition of M_{12}, we have

$$\hat{U}^T M_{12} U = \Sigma + \delta_{\mathbf{u}} + O(\mu), \quad \hat{U}^T M_{12} V = \delta_{\mathbf{u}} + O(\mu), \tag{4.13a}$$

$$\hat{V}^T M_{12} U = \delta_{\mathbf{u}} + O(\mu), \quad \hat{V}^T M_{12} V = \delta_{\mathbf{u}} + O(\mu). \tag{4.13b}$$

For the M_{22} block, we have from (4.9) and (4.6) that

$$U^T M_{22} U = -U^T D_{\mathcal{B}} U + \delta_{\mathbf{u}} = O(\mu) + \delta_{\mathbf{u}}, \tag{4.14a}$$

$$U^T M_{22} V = O(\mu) + \delta_{\mathbf{u}}, \quad V^T M_{22} U = O(\mu) + \delta_{\mathbf{u}}, \tag{4.14b}$$

$$V^T M_{22} V = -V^T D_{\mathcal{B}} V + \delta_{\mathbf{u}} = \tilde{M}_{VV} + \delta_{\mathbf{u}}, \tag{4.14c}$$

where $\tilde{M}_{VV} \overset{\text{def}}{=} -V^T D_{\mathcal{B}} V$ has all its singular values of size $\Theta(\mu)$, so that

$$\tilde{M}_{VV}^{-1} = \Theta(\mu^{-1}). \tag{4.15}$$

Using these estimates together with (4.10), we can rewrite (4.8) as

$$\left\{ \begin{bmatrix} Q & 0 \\ 0 & \tilde{M}_{VV} \end{bmatrix} + \begin{bmatrix} \hat{E}_{11} & \hat{E}_{12} \\ \hat{E}_{21} & \hat{E}_{22} \end{bmatrix} \right\} \left[\begin{array}{c} w_{\hat{U}} \\ w_{\hat{V}} \\ y_U \\ \hline y_V \end{array} \right] = \left[\begin{array}{c} \hat{U}^T r_1 \\ \hat{V}^T r_1 \\ \Sigma \hat{U}^T r_3 + U^T r_4 \\ \hline V^T r_4 \end{array} \right], \tag{4.16}$$

where

$$Q = \begin{bmatrix} \hat{U}^T \mathcal{L}_{zz}(z^*, \lambda^*) \hat{U} & \hat{U}^T \mathcal{L}_{zz}(z^*, \lambda^*) \hat{V} & \Sigma \\ \hat{V}^T \mathcal{L}_{zz}(z^*, \lambda^*) \hat{U} & \hat{V}^T \mathcal{L}_{zz}(z^*, \lambda^*) \hat{V} & 0 \\ \Sigma & 0 & 0 \end{bmatrix} \tag{4.17}$$

$$+ \begin{bmatrix} \delta_{\mathbf{u}}/\mu + O(\mu) & \delta_{\mathbf{u}}/\mu + O(\mu) & \delta_{\mathbf{u}} + O(\mu) \\ \delta_{\mathbf{u}}/\mu + O(\mu) & \delta_{\mathbf{u}}/\mu + O(\mu) & 0 \\ \delta_{\mathbf{u}} + O(\mu) & 0 & 0 \end{bmatrix}$$

$$\overset{\text{def}}{=} \begin{bmatrix} N_{UU} & N_{UV} & \bar{\Sigma}_1 \\ N_{VU} & N_{VV} & 0 \\ \bar{\Sigma}_2 & 0 & 0 \end{bmatrix}, \tag{4.18}$$

while

$$\hat{E}_{11} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \delta_{\mathbf{u}} + O(\mu) \\ 0 & \delta_{\mathbf{u}} + O(\mu) & \delta_{\mathbf{u}} + O(\mu) \end{bmatrix}, \tag{4.19}$$

and

$$\hat{E}_{12}, \hat{E}_{21} = \delta_{\mathbf{u}} + O(\mu) = O(\mu), \quad \hat{E}_{22} = \delta_{\mathbf{u}}. \tag{4.20}$$

From the estimates (4.7b) and (4.7c), we have for β sufficiently large that the diagonal blocks of Q are all invertible and that their eigenvalues do not depart too much in a relative sense from the eigenvalues of Σ and $\hat{V}^T\mathcal{L}_{zz}(z^*,\lambda^*)\hat{V}$, as appropriate. In fact, we have for sufficiently large β that

$$\|\bar{\Sigma}_1\| \le 2\|\Sigma\|, \quad \|\bar{\Sigma}_1^{-1}\| \le 2\|\Sigma^{-1}\|, \tag{4.21a}$$

$$\|\bar{\Sigma}_2\| \le 2\|\Sigma\|, \quad \|\bar{\Sigma}_2^{-1}\| \le 2\|\Sigma^{-1}\|, \tag{4.21b}$$

$$\|N_{VV}\| \le 2\|\hat{V}^T\mathcal{L}_{zz}(z^*,\lambda^*)\hat{V}\|, \quad \|N_{VV}^{-1}\| \le 2\|(\hat{V}^T\mathcal{L}_{zz}(z^*,\lambda^*)\hat{V})^{-1}\|. \tag{4.21c}$$

Note, too, that because of Lipschitz continuity of $\mathcal{L}_{zz}$ and compactness of $\mathcal{S}$, and the bounds (4.7a), the norms of N_{UU}, N_{UV}, N_{VU}, N_{VV}, and Σ are all $O(1)$. Hence the matrix Q is itself invertible, and we have

$$(4.22)\ Q^{-1} = \begin{bmatrix} 0 & 0 & \bar{\Sigma}_2^{-1} \\ 0 & N_{VV}^{-1} & -N_{VV}^{-1}N_{VU}\bar{\Sigma}_2^{-1} \\ \bar{\Sigma}_1^{-1} & -\bar{\Sigma}_1^{-1}N_{UV}N_{VV}^{-1} & -\bar{\Sigma}_1^{-1}(N_{UU} - N_{UV}N_{VV}^{-1}N_{VU})\bar{\Sigma}_2^{-1} \end{bmatrix}.$$

(This claim can be verified easily by forming the product QQ^{-1} using the partitions (4.18) and (4.22).) Noting that

$$(Q + \hat{E}_{11})^{-1} = (I + Q^{-1}\hat{E}_{11})^{-1}Q^{-1}, \tag{4.23}$$

we examine the size of $Q^{-1}\hat{E}_{11}$. Note first from (4.7b) and (4.7c) together with that

$$\|\bar{\Sigma}_1^{-1}\| \le \frac{2}{\beta}(\mathbf{u}/\mu)^{-1}, \quad \|\bar{\Sigma}_2^{-1}\| \le \frac{2}{\beta}(\mathbf{u}/\mu)^{-1}, \quad \|N_{VV}^{-1}\| \le \frac{2}{\beta}(\mathbf{u}/\mu)^{-1}, \tag{4.24a}$$

$$\|\bar{\Sigma}_1^{-1}\| \le \frac{2}{\beta}\mu^{-1/3}, \quad \|\bar{\Sigma}_2^{-1}\| \le \frac{2}{\beta}\mu^{-1/3}, \quad \|N_{VV}^{-1}\| \le \frac{2}{\beta}\mu^{-1/3}. \tag{4.24b}$$

By forming the product of (4.22) with (4.19) and using the bounds in (4.24), we can show that the norm of $Q^{-1}\hat{E}_{11}$ can be made arbitrarily small by choosing β sufficiently large. The $(3,3)$ block of $Q^{-1}\hat{E}_{11}$, for instance, has the form

$$-\bar{\Sigma}_1^{-1}N_{UV}N_{VV}^{-1}(\delta_{\mathbf{u}} + O(\mu)) + \bar{\Sigma}_1^{-1}(N_{UU} - N_{UV}N_{VV}^{-1}N_{VU})\bar{\Sigma}_2^{-1}(\delta_{\mathbf{u}} + O(\mu)),$$

whose norm is bounded by a quantity of the form

$$\left(\|\bar{\Sigma}_1^{-1}\|\,\|N_{VV}^{-1}\| + \|\bar{\Sigma}_1^{-1}\|\,\|\bar{\Sigma}_2^{-1}\|\,\|N_{VV}^{-1}\| + \|\bar{\Sigma}_1^{-1}\|\,\|\bar{\Sigma}_2^{-1}\|\right)\left((\delta_{\mathbf{u}}/\mu)\mu + O(\mu)\right),$$

which in turn because of (4.24) is bounded by the following quantity, for some C sufficiently large and independent of μ and β:

$$C\left(\frac{1}{\beta^2}\mu^{2/3} + \frac{1}{\beta^3}\mu^{1/3}\right) + C\left(\frac{1}{\beta^2}\mu^{1/3} + \frac{1}{\beta^3}\right),$$

which can be made as small as we like by choosing β large enough. Therefore, by appropriate choice of β, we have that $\|(I + Q^{-1}\hat{E}_{11})^{-1}\| \le 2$, and so from (4.23) we have

$$\|(Q + \hat{E}_{11})^{-1}\| = 2\|Q^{-1}\|.$$

Our conclusion is that for appropriate choice of β in (4.7), the inverse of the $(1,1)$ block of the matrix in (4.16) can be bounded in terms of $\|Q^{-1}\|$, which because of

(4.21) and (4.22) can in turn be bounded by a finite quantity that depends only on the problem data and not on μ.

Returning to (4.16), we have that

$$\begin{aligned} \begin{bmatrix} w_{\hat U} \\ w_{\hat V} \\ y_U \end{bmatrix} &= -(Q+\hat E_{11})^{-1}\hat E_{12} y_V + (Q+\hat E_{11})^{-1}\begin{bmatrix} \hat U^T r_1 \\ \hat V^T r_1 \\ \Sigma \hat U^T r_3 + U^T r_4 \end{bmatrix} \\ &= O(\|\hat E_{12}\|\|y_V\|) + O(\|r_1\| + \|r_3\| + \|r_4\|) \\ &= O(\mu)\|y_V\| + O(\|r_1\| + \|r_3\| + \|r_4\|). \end{aligned} \tag{4.25}$$

Meanwhile, for the second block row of (4.16), we obtain

$$y_V = -(\tilde M_{VV} + \hat E_{22})^{-1}\hat E_{21}\begin{bmatrix} w_{\hat U} \\ w_{\hat V} \\ y_U \end{bmatrix} + (\tilde M_{VV} + \hat E_{22})^{-1} V^T r_4. \tag{4.26}$$

Since from (4.15), (4.20), and (4.7a), we have

$$(\tilde M_{VV} + \hat E_{22})^{-1} = (I + \tilde M_{VV}^{-1}\hat E_{22})^{-1}\tilde M_{VV}^{-1} = (I + \delta_{\mathbf{u}}/\mu)\tilde M_{VV}^{-1} = O(\mu^{-1}),$$

it follows from (4.26) and (4.20) that

$$y_V = O(\mu^{-1})O(\mu)\left\|\begin{bmatrix} w_{\hat U} \\ w_{\hat V} \\ y_U \end{bmatrix}\right\| + O(\mu^{-1})O(\|r_4\|).$$

By substituting from (4.25), we obtain

$$\|y_V\| = O(\mu)\|y_V\| + O(\|r_1\| + \|r_3\| + \|r_4\|) + O(\|r_4\|/\mu),$$

and therefore

$$\|y_V\| = O(\|r_1\| + \|r_3\| + \|r_4\|/\mu),$$

as claimed. The estimate for $(w_{\hat U}, w_{\hat V}, y_U)$ is obtained by substituting into (4.25). $\square$

The conditions (4.7) need a little explanation. For the typical value $\mathbf{u} = 10^{-16}$, the minimum value of the quantity $\max(\mu^{1/3}, \mathbf{u}/\mu)$ is 10^{-4}, achieved at μ^{-12}. Moreover, we have $\max(\mu^{1/3}, \mathbf{u}/\mu) \le 10^{-2}$ only for μ in the range $[10^{-14}, 10^{-6}]$. It would seem, then, that the problem would need to be quite well conditioned for (4.7b) and (4.7c) to hold and that μ may have to become quite small before the results apply. We note, however, that the purpose of the bounds (4.7b) and (4.7c) is to ensure that inverse of $Q + \hat E_{11}$ can be bounded independently of μ, and that for this purpose they are quite conservative. That is, we would expect to find that $\|(Q + \hat E_{11})^{-1}\|$ is not too much larger than the norm of the inverse of the correspinding exact matrix (the first term on the right-hand side of (4.17)) for μ not much less than the smallest eigenvalues of Σ and $\hat V^T \mathcal{L}_{zz}(z^*, \lambda^*)\hat V$.

Their requirement that $\mathbf{u}/\mu$ and μ both be small in (4.7) may not seem to sit well with expressions such as $O(\mu)$ and $O(\mu^2)$, which crop up repeatedly in the analysis and which assert that certain bounds hold "for all sufficiently small μ." As noted in the preceding paragraph, this requirement implies that the analysis holds for μ in a certain range, or "window," of values. Similar windows are used the analysis of

S. Wright [25, 28, 31], and M. H. Wright [24], and numerical experience indicates that such a window does indeed exist in most practical cases. We expect the same to be true of the problem and algorithms discussed in this paper.

At this point, we assemble the assumptions that are made in the remainder of the paper into a single catch-all assumption.

ASSUMPTION 4.1.

(a) *z^* is a solution of (1.1), so that the condition (1.3) holds. The MFCQ condition (2.5), the second-order conditions (2.6), (2.7), and the SC condition (1.5) are satisfied at this solution. The current iterate (z, λ, s) of the PDIP algorithm satisfies the conditions (3.11), and the right-hand side modification t satisfies (3.7).*

(b) *The quantities μ, $\mathbf{u}$ (2.10), $\mathcal{L}_{zz}(z^*, \lambda^*)$, Σ, and $\hat{V}$ (2.13) satisfy the conditions (4.7).*

Our next result considers a perturbed form of the system (4.1), with a general right-hand side. By eliminating one component to obtain the form (4.3), we can apply Theorem 4.1 to obtain estimates of the dependence of the solution on the right-hand side components.

THEOREM 4.2. *Suppose that Assumption 4.1 holds. Consider the linear system*

$$\begin{bmatrix} \mathcal{L}_{zz}(z,\lambda) + E_{11} & \nabla g_{\mathcal{B}}(z) + E_{12} & \nabla g_{\mathcal{N}}(z) + E_{13} \\ \nabla g_{\mathcal{B}}(z)^T + E_{21} & -D_{\mathcal{B}} + E_{22} & E_{23} \\ \nabla g_{\mathcal{N}}(z)^T + E_{31} & E_{32} & -D_{\mathcal{N}} + E_{33} \end{bmatrix} \begin{bmatrix} w \\ y \\ q \end{bmatrix} = \begin{bmatrix} r_5 \\ \nabla g_{\mathcal{B}}(z^*)^T r_6 + r_7 \\ r_8 \end{bmatrix}, \tag{4.27}$$

where

$$E_{11} = \delta_{\mathbf{u}}/\mu, \qquad E_{33} = \delta_{\mathbf{u}}/\mu^2, \tag{4.28a}$$

$$E_{12}, E_{21}, E_{22} = \delta_{\mathbf{u}}, \qquad E_{13}, E_{31}, E_{23}, E_{32} = \delta_{\mathbf{u}}/\mu. \tag{4.28b}$$

Then the step (w, y, q) satisfies the following estimates:

$$\begin{aligned} (U^T y, w) &= O(\|r_5\| + \|r_6\| + \|r_7\| + \mu\|r_8\|), \\ V^T y &= O(\|r_5\| + \|r_6\| + \|r_7\|/\mu + (\delta_{\mathbf{u}}/\mu + O(\mu))\|r_8\|), \\ q &= O(\mu)\left[\|r_5\| + \|r_6\| + \|r_8\|\right] + (\delta_{\mathbf{u}}/\mu + O(\mu))\|r_7\|. \end{aligned}$$

Proof. Note first that Assumption 4.1 ensures that the conditions of Lemmas 3.1 and 3.2 are satisfied. Hence, from Lemma 3.2, we have that the diagonal elements of $D_{\mathcal{B}}$ in (4.2) have size $\Theta(\mu)$, while the diagonal elements of $D_{\mathcal{N}}$ have size $\Theta(\mu^{-1})$.

From the assumed bound (4.28a) on the size of E_{33}, we have that

$$\begin{aligned} &(-D_{\mathcal{N}} + E_{33})^{-1} \\ &= -(I - D_{\mathcal{N}}^{-1} E_{33})^{-1} D_{\mathcal{N}}^{-1} = (I + O(\mu)\delta_{\mathbf{u}}/\mu^2) O(\mu) = O(\mu). \end{aligned} \tag{4.29}$$

By eliminating q from (4.27), we obtain the reduced system

$$\begin{bmatrix} H(z,\lambda) + \tilde{E}_{11} & \nabla g_{\mathcal{B}}(z) + \tilde{E}_{12} \\ \nabla g_{\mathcal{B}}(z)^T + \tilde{E}_{21} & -D_{\mathcal{B}} + \tilde{E}_{22} \end{bmatrix} \begin{bmatrix} w \\ y \end{bmatrix} = \begin{bmatrix} r_5 + O(\mu)\|r_8\| \\ \nabla g_{\mathcal{B}}(z^*)^T r_6 + r_7 + \delta_{\mathbf{u}}\|r_8\| \end{bmatrix},$$

where from (4.7) and (4.4), we obtain

$$\begin{aligned}
\tilde E_{11} &= E_{11} - (\nabla g_{\mathcal{N}}(z) + E_{13})(-D_{\mathcal{N}} + E_{33})^{-1}(\nabla g_{\mathcal{N}}(z)^T + E_{31}) - \nabla g_{\mathcal{N}}(z) D_{\mathcal{N}}^{-1} \nabla g_{\mathcal{N}}(z)^T \\
&= \delta_{\mathbf{u}}/\mu + O(\mu), \\
\tilde E_{12} &= E_{12} - (\nabla g_{\mathcal{N}}(z) + E_{13})(-D_{\mathcal{N}} + E_{33})^{-1} E_{32} = \delta_{\mathbf{u}} + O(1)O(\mu)\delta_{\mathbf{u}}/\mu = \delta_{\mathbf{u}}, \\
\tilde E_{21} &= E_{21} - E_{23}(-D_{\mathcal{N}} + E_{33})^{-1}(\nabla g_{\mathcal{N}}(z)^T + E_{31}) = \delta_{\mathbf{u}} + (\delta_{\mathbf{u}}/\mu)O(\mu)O(1) = \delta_{\mathbf{u}}, \\
\tilde E_{22} &= E_{22} - E_{23}(-D_{\mathcal{N}} + E_{33})^{-1} E_{32} = \delta_{\mathbf{u}} + (\delta_{\mathbf{u}}/\mu)^2 O(\mu) = \delta_{\mathbf{u}}.
\end{aligned}$$

These perturbation matrices satisfy the assumptions of Theorem 4.1, which can be applied to give

$$(U^T y, \hat V^T w, \hat U^T w) = O(\|r_5\| + \|r_6\| + \|r_7\| + \mu\|r_8\|), \tag{4.30a}$$

$$V^T y = O(\|r_5\| + \|r_6\| + \|r_7\|/\mu) + (\delta_{\mathbf{u}}/\mu + O(\mu))\|r_8\|. \tag{4.30b}$$

From the last block row of (4.27), and using (4.7), (4.29), (4.30), we obtain

$$\begin{aligned}
q &= (-D_{\mathcal{N}} + E_{33})^{-1}\left[r_8 - (\nabla g_{\mathcal{N}}(z)^T + E_{31})w - E_{32}y\right] \\
&= O(\mu)\left[\|r_8\| + \|w\| + (\delta_{\mathbf{u}}/\mu)\|y\|\right] \\
&= O(\mu)\left[\|r_5\| + \|r_6\| + \|r_7\| + \|r_8\|\right] + \\
&\qquad \delta_{\mathbf{u}}\left[\|r_5\| + \|r_6\| + \|r_7\|/\mu + (\delta_{\mathbf{u}}/\mu + O(\mu))\|r_8\|\right] \\
&= O(\mu)\left[\|r_5\| + \|r_6\| + \|r_8\|\right] + (\delta_{\mathbf{u}}/\mu + O(\mu))\|r_7\|.
\end{aligned}$$

□

An estimate for the solution of the exact system (3.8) follows almost immediately from this result. This is the key technical result used by Ralph and Wright [23, 22] to prove superlinear convergence of PDIP algorithms for convex programming problems. The result below, however, does not require a convexity assumption.

COROLLARY 4.3. *Suppose that Assumption 4.1(a) holds. Then the (exact) solution* $(\Delta z, \Delta\lambda, \Delta s)$ *of the system (3.8) satisfies*

$$(\Delta z, \Delta\lambda, \Delta s) = O(\mu). \tag{4.31}$$

Proof. Note first that Assumption 4.1(b) holds trivially in this case for μ sufficiently small, because our assumption of exact computations is equivalent to setting $\mathbf{u} = 0$. We prove the result by identifying the system (4.1) with (4.27) and then applying Theorem 4.2.

For the right-hand side, we note first that, because of smoothness of g, Taylor's theorem, the definition (2.1) of $\mathcal{B}$, and Theorem 3.3,

$$\begin{aligned}
g_{\mathcal{B}}(z) &= g_{\mathcal{B}}(z^*) + \nabla g_{\mathcal{B}}(z^*)^T(z - z^*) + O(\|z - z^*\|^2) \\
&= \nabla g_{\mathcal{B}}(z^*)^T(z - z^*) + O(\mu^2).
\end{aligned} \tag{4.32}$$

We now identify the right-hand side of (4.1) with (4.27) by setting

$$\begin{aligned}
r_5 &= -\mathcal{L}_z(z, \lambda), \\
r_6 &= (z - z^*), \\
r_7 &= -\nabla g_{\mathcal{B}}(z^*)^T(z - z^*) - g_{\mathcal{B}}(z) + \Lambda_{\mathcal{B}}^{-1} t_{\mathcal{B}}, \\
r_8 &= -g_{\mathcal{N}}(z) + \Lambda_{\mathcal{N}}^{-1} t_{\mathcal{N}}.
\end{aligned}$$

The sizes of these vectors can be estimated by using (3.11), Lemma 3.2, (4.32), Theorem 3.3, and the assumption (3.7) on the size of t to obtain

$$r_5 = O(\mu), \quad r_6 = O(\mu), \quad r_7 = O(\mu^2), \quad r_8 = O(1). \tag{4.33}$$

(By choosing r_6 and r_7 in this way, we ensure that the terms involving $\|r_7\|/\mu$ in the estimates of the solution components in Theorem 4.2 are not grossly larger than the other terms in these expressions.) We complete the identification of (4.1) with (4.27) by setting all the perturbation matrices $E_{11}, E_{12}, \dots, E_{33}$ to zero and by identifying the solution vector components Δz, $\Delta\lambda_{\mathcal{B}}$, and $\Delta\lambda_{\mathcal{N}}$ with w, y, and q, respectively. By directly applying Theorem 4.2, substituting the estimates (4.33), and setting $\delta_{\mathbf{u}} = 0$ (since we are assuming exact computations), we have that

$$(U^T\Delta\lambda_{\mathcal{B}}, \Delta z) = O(\mu), \quad V^T\Delta\lambda = O(\mu), \quad \Delta\lambda_{\mathcal{N}} = O(\mu),$$

so that $(\Delta z, \Delta\lambda) = O(\mu)$.

To show that the remaining solution component Δs of (3.8) is also of size $O(\mu)$, we write the second block row in (3.8) as

$$\Delta s = -(g(z) + s) - \nabla g(z)^T \Delta z,$$

from which the desired estimate follows immediately by substituting from (3.11b) and $\Delta z = O(\mu)$. □

The next result uses Theorem 4.2 to compare perturbed and exact solutions of the system of the system (4.1).

Corollary 4.4. *Suppose that Assumption 4.1 holds. Let (w, y, q) be obtained from the following perturbed version of (3.9):*

$$\begin{bmatrix} \mathcal{L}_{zz}(z,\lambda) + E_{11} & \nabla g_{\mathcal{B}}(z) + E_{12} & \nabla g_{\mathcal{N}}(z) + E_{13} \\ \nabla g_{\mathcal{B}}(z)^T + E_{21} & -D_{\mathcal{B}} + E_{22} & E_{23} \\ \nabla g_{\mathcal{N}}(z)^T + E_{31} & E_{32} & -D_{\mathcal{N}} + E_{33} \end{bmatrix} \begin{bmatrix} w \\ y \\ q \end{bmatrix} = \begin{bmatrix} -\mathcal{L}_z(z,\lambda) + f_1 \\ -g_{\mathcal{B}}(z) + \Lambda_{\mathcal{B}}^{-1} t_{\mathcal{B}} + f_2 \\ -g_{\mathcal{N}}(z) + \Lambda_{\mathcal{N}}^{-1} t_{\mathcal{N}} + f_3 \end{bmatrix}, \tag{4.34}$$

where E_{ij}, $i, j = 1, 2, 3$, satisfy the conditions (4.28) and f_1, f_2, and f_3 are all of size $\delta_{\mathbf{u}}$. Then if $(\Delta z, \Delta\lambda, \Delta s)$ is the (exact) solution of the system (3.8), we have

$$\begin{aligned} (\Delta z - w, U^T(\Delta\lambda_{\mathcal{B}} - y)) &= \delta_{\mathbf{u}}, \\ V^T(\Delta\lambda_{\mathcal{B}} - y) &= \delta_{\mathbf{u}}/\mu, \\ \Delta\lambda_{\mathcal{N}} - q &= \delta_{\mathbf{u}}. \end{aligned}$$

Proof. By combining (4.34) with (4.1), we obtain

$$\begin{bmatrix} \mathcal{L}_{zz}(z,\lambda) + E_{11} & \nabla g_{\mathcal{B}}(z) + E_{12} & \nabla g_{\mathcal{N}}(z) + E_{13} \\ \nabla g_{\mathcal{B}}(z)^T + E_{21} & -D_{\mathcal{B}} + E_{22} & E_{23} \\ \nabla g_{\mathcal{N}}(z)^T + E_{31} & E_{32} & -D_{\mathcal{N}} + E_{33} \end{bmatrix} \begin{bmatrix} w - \Delta z \\ y - \Delta\lambda_{\mathcal{B}} \\ q - \Delta\lambda_{\mathcal{N}} \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} - \begin{bmatrix} E_{11} & E_{12} & E_{13} \\ E_{21} & E_{22} & E_{23} \\ E_{31} & E_{32} & E_{33} \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta\lambda_{\mathcal{B}} \\ \Delta\lambda_{\mathcal{N}} \end{bmatrix}. \tag{4.35}$$

From the bounds on the perturbations E in (4.28) and the estimate (4.31) of the exact step, we have for the right-hand side of this expression that

$$\begin{bmatrix} r_5 \\ r_7 \\ r_8 \end{bmatrix} \stackrel{\text{def}}{=} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} - \begin{bmatrix} E_{11} & E_{12} & E_{13} \\ E_{21} & E_{22} & E_{23} \\ E_{31} & E_{32} & E_{33} \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta\lambda_{\mathcal{B}} \\ \Delta\lambda_{\mathcal{N}} \end{bmatrix} = \begin{bmatrix} \delta_{\mathbf{u}} \\ \delta_{\mathbf{u}} \\ \delta_{\mathbf{u}}/\mu \end{bmatrix}.$$

Using these estimates, we can simply apply Theorem 4.2 to (4.35) (with $r_6 = 0$) to obtain the result. ☐

We emphasize that the conditions (3.11), and in particular (3.11c), are critical to the results of this and all the remaining sections of the paper. These conditions enable the estimates (3.16) of Lemma 3.2, which in turn enable us to assert that the diagonals of $D_{\mathcal{B}}$ all have size $\Theta(\mu)$ while those of $D_{\mathcal{N}}$ all have size $\Theta(\mu^{-1})$. This neat classification of the diagonals of D into two categories drives all the subsequent analysis. The motivation for conditions like (3.11) in the literature for path-following methods (with exact steps) is not unrelated: It allows us to obtain bounds on the steps and to show that we can move a significant distance along this direction while ensuring that (3.11) continues to be satisfied at the new iterate. (See, for example, [27, Chapters 5 and 6] and its bibliography for the case of linear programming and [32, 23, 22] for the case of nonlinear convex programming.) In the analysis above, we obtain bounds on the *errors* (rather than the steps themselves) when perturbation terms of a certain structure appear in the matrix and right-hand side.

Many practical implementations of path-following methods for linear programming do not explicitly check that the condition (3.11c) is satisfied by the calculated iterates (see, for example, [20] and [5]). However, the heuristics for "stepping back" from the boundary of the nonnegative orthant by a small but significant quantity are motivated by this condition, and it is observed to hold in practice on all but the most recalcitrant problems.

5. The Condensed System.

Here we consider an algorithm in which the condensed linear system (3.10) is formed and solved to obtain Δz, and the remaining step components $\Delta\lambda$ and Δs are recovered from (3.8). We obtain expressions for the errors in the calculated step $(\widehat{\Delta z}, \widehat{\Delta\lambda}, \widehat{\Delta s})$ and discuss the effects of these errors on certain measures of step quality. We also derive conditions under which the Cholesky factorization applied to (3.10) is guaranteed to run to completion.

Formally, the complete procedure can be described as follows:

```
procedure condensed
given the current iterate $(z, \lambda, s)$
    form the coefficient matrix and right-hand side for (3.10);
    solve (3.10) using a backward stable algorithm to obtain $\Delta z$;
    set $\Delta\lambda = D^{-1}[g(z) - \Lambda^{-1}t + \nabla g(z)^T \Delta z]$;
    set $\Delta s = -(g(z) + s) - \nabla g(z)^T \Delta z$.
```

We have used the definition (4.2) of the matrix D. For convenience, we restate the system (3.10) here as follows:

$$\left[\mathcal{L}_{zz}(z,\lambda) + \nabla g(z) D^{-1} \nabla g(z)^T\right] \Delta z = -\mathcal{L}_z(z,\lambda) - \nabla g(z) D^{-1}[g(z) - \Lambda^{-1}t]. \tag{5.1}$$

Note that this procedure requires evaluation of $D^{-1} = S^{-1}\Lambda$, rather than D itself.

5.1. Quantifying the Errors. When implemented in finite-precision arithmetic, solution of (5.1) gives rise to errors of three types:

- cancellation in evaluation of the matrix and right-hand side;
- roundoff errors in evaluation of the matrix and right-hand side;
- roundoff errors that accumulate during the process of factoring the matrix and using triangular substitutions to obtain the solution.

Cancellation may be an issue in the evaluation of the nonlinear functions $\mathcal{L}_{zz}(z,\lambda)$, $\mathcal{L}_z(z,\lambda)$, $g(z)$, and $\nabla g(z)$, because intermediate terms computed during the additive evaluation so these quantities may exceed the size of the final result (see Golub and Van Loan [12, p. 61]). The intermediate terms generally contain rounding error (which occurs whenever real numbers are represented in finite precision). Intermediate quantities computed during the evaluation of these functions may contain rounding error, which occurs when real numbers are represented in finite precision. Cancellation becomes a significant phenomenon whenever we take a difference of two nearly equal quantities, since the error in the computed result due to roundoff in the two arguments may be large relative to the size of the result. If, as we can reasonably assume, intermediate quantities in the calculations of our right-hand sides remain bounded, the absolute size of the errors in the result is $\delta_{\mathbf{u}}$. In the case of $\mathcal{L}_z(z,\lambda)$ and $g_{\mathcal{B}}(z)$, the final result in exact arithmetic has size $O(\mu)$, so that the error $\delta_{\mathbf{u}}$ takes on a large relative significance for small values of μ. This fact causes the error bound in some components of the solution to be larger than in others, as we see in (5.6c) below. In summary, the computed versions of the quantities discussed above differ from their exact values in the following way:

$$\text{computed } \mathcal{L}_{zz}(z,\lambda) \leftarrow \mathcal{L}_{zz}(z,\lambda) + \bar{F}, \tag{5.2a}$$

$$\text{computed } \mathcal{L}_{z}(z,\lambda) \leftarrow \mathcal{L}_{z}(z,\lambda) + \bar{f}, \tag{5.2b}$$

$$\text{computed } \nabla g(z) \leftarrow \nabla g(z) + F = \begin{bmatrix} \nabla g_{\mathcal{B}}(z) \\ \nabla g_{\mathcal{N}}(z) \end{bmatrix} + \begin{bmatrix} F_{\mathcal{B}} \\ F_{\mathcal{N}} \end{bmatrix}, \tag{5.2c}$$

$$\text{computed } g(z) \leftarrow g(z) + f = \begin{bmatrix} g_{\mathcal{B}}(z) \\ g_{\mathcal{N}}(z) \end{bmatrix} + \begin{bmatrix} f_{\mathcal{B}} \\ f_{\mathcal{N}} \end{bmatrix}, \tag{5.2d}$$

where $\bar{F}$, $\bar{f}$, F, and f are all of size $\delta_{\mathbf{u}}$. Earlier discussion of cancellation in similar contexts can be found in the papers of S. Wright [25, 28, 31] and M. H. Wright[24].

The second source of error is evaluation of the matrix D^{-1}. From the model (2.10) of floating-point arithmetic and the estimates (3.16) of Lemma 3.2, we have that

$$\text{computed } D_{\mathcal{B}}^{-1} \leftarrow (D_{\mathcal{B}} + G_{\mathcal{B}})^{-1}, \quad G_{\mathcal{B}} = \mu\delta_{\mathbf{u}}, \tag{5.3a}$$

$$\text{computed } D_{\mathcal{N}}^{-1} \leftarrow (D_{\mathcal{N}} + G_{\mathcal{N}})^{-1}, \quad G_{\mathcal{N}} = \delta_{\mathbf{u}}/\mu, \tag{5.3b}$$

where $G_{\mathcal{B}}$ and $G_{\mathcal{N}}$ are both diagonal matrices that can be composed into a single diagonal matrix G.

Third, we account for the error in forming the matrix and right-hand side of (5.1) from the computed quantities described in the last two paragraphs. Since we are now dealing with floating-point numbers, the model (2.10) applies; that is, any additional errors that arise during the combination of these floating-point quantities have size $\mathbf{u}$ relative to the size of the result of the calculation. Since the norm of the coefficient matrix is of size $O(\mu^{-1})$ while the right-hand side has size $O(1)$ (see (3.11)), we represent these errors by a matrix $\hat{F}$ of size $\delta_{\mathbf{u}}/\mu$ and a vector $\hat{f}$ f size $\delta_{\mathbf{u}}$.

Finally, we account for the error that arises in the application of a backward-stable method to solve (5.1). Specifically, we assume that the method yields a computed solution that is the exact solution of a nearby problem whose data contains relative perturbations of size $\mathbf{u}$. The absolute sizes of these terms would therefore be $\delta_{\mathbf{u}}/\mu$ in the case of the matrix and $\delta_{\mathbf{u}}$ in the case of the right-hand side. Since these errors are the same size as those discussed in the preceding paragraph, we incorporate them into the matrix $\hat{F}$ and the vector $\hat{f}$.

Summarizing, we find that the computed solution $\widehat{\Delta z}$ of (5.1) satisfies the following system:

$$\begin{aligned}(5.4)\qquad &\left[\mathcal{L}_{zz}(z,\lambda) + \bar{F} + (\nabla g(z) + F)(D+G)^{-1}(\nabla g(z) + F)^T + \hat{F}\right]\widehat{\Delta z} \\ &= -\mathcal{L}_z(z,\lambda) - \bar{f} - (\nabla g(z) + F)(D+G)^{-1}[g(z) + f - \Lambda^{-1}t] + \hat{f},\end{aligned}$$

where the perturbation terms $\bar{F}$, F, $\hat{F}$, G, $\bar{f}$, $\hat{f}$, and f are described in the paragraphs above. By "unfolding" this system and using the partitioning of F, G, and f defined in (5.2) and (5.3), we find that $\widehat{\Delta z}$ also satisfies the following system, for some vectors y and q:

$$(5.5)\qquad \begin{bmatrix} \mathcal{L}_{zz}(z,\lambda) + \bar{F} + \hat{F} & \nabla g_{\mathcal{B}}(z) + F_{\mathcal{B}} & \nabla g_{\mathcal{N}}(z) + F_{\mathcal{N}} \\ \nabla g_{\mathcal{B}}(z)^T + F_{\mathcal{B}}^T & -D_{\mathcal{B}} - G_{\mathcal{B}} & 0 \\ \nabla g_{\mathcal{N}}(z)^T + F_{\mathcal{N}}^T & 0 & -D_{\mathcal{N}} - G_{\mathcal{N}} \end{bmatrix} \begin{bmatrix} \widehat{\Delta z} \\ y \\ q \end{bmatrix} = \begin{bmatrix} -\mathcal{L}_z(z,\lambda) - \bar{f} + \hat{f} \\ -g_{\mathcal{B}}(z) + \Lambda_{\mathcal{B}}^{-1}t_{\mathcal{B}} - f_{\mathcal{B}} \\ -g_{\mathcal{N}}(z) + \Lambda_{\mathcal{N}}^{-1}t_{\mathcal{N}} - f_{\mathcal{N}} \end{bmatrix}.$$

This system has precisely the form of (4.34) (in particular, the perturbation matrices satisfy the appropriate bounds). Hence, by a direct application of Corollary 4.4, we conclude that

$$(5.6a)\qquad \Delta z - \widehat{\Delta z} = \delta_{\mathbf{u}},$$

$$(5.6b)\qquad U^T(\Delta\lambda_{\mathcal{B}} - y) = \delta_{\mathbf{u}},$$

$$(5.6c)\qquad V^T(\Delta\lambda_{\mathcal{B}} - y) = \delta_{\mathbf{u}}/\mu.$$

We return now to the recovery of the remaining solution components $\widehat{\Delta\lambda}$ and $\widehat{\Delta s}$ in the procedure **condensed**. We have from Assumption 4.1 together with (5.2), (5.3a), (4.31), (5.6a), (4.2), Lemma 3.2, (3.11b), and (4.7) that

$$(5.7a)\qquad g_{\mathcal{B}}(z) = r_g(z,s)_{\mathcal{B}} - s_{\mathcal{B}} = O(\mu), \qquad \Lambda_{\mathcal{B}}^{-1} = \Theta(1),$$

$$(5.7b)\qquad \widehat{\Delta z} = \Delta z + \delta_{\mathbf{u}} = O(\mu), \qquad (D_{\mathcal{B}} + G_{\mathcal{B}})^{-1} = \Theta(\mu^{-1}).$$

Since $t = O(\mu^2)$, we have from our model (2.10) that the floating-point version of the calculation of $\widehat{\Delta\lambda}_{\mathcal{B}}$ in the procedure **condensed** satisfies the following:

$$\widehat{\Delta\lambda}_{\mathcal{B}} = (D_{\mathcal{B}} + G_{\mathcal{B}})^{-1}\left[g_{\mathcal{B}}(z) + f_{\mathcal{B}} - \Lambda_{\mathcal{B}}^{-1}t_{\mathcal{B}} + (\nabla g_{\mathcal{B}}(z) + F_{\mathcal{B}})^T\widehat{\Delta z} + \mu\delta_{\mathbf{u}}\right] + \delta_{\mathbf{u}}.$$

(The final term $\delta_{\mathbf{u}}$ arises from (2.10) because our best estimate of the quantity in the brackets at this point of the analysis is $O(\mu)$, so from (5.7b) the result has size $O(1)$.) Meanwhile, we have from the second block row of (5.5) that

$$y = (D_{\mathcal{B}} + G_{\mathcal{B}})^{-1}\left[g_{\mathcal{B}}(z) + f_{\mathcal{B}} - \Lambda_{\mathcal{B}}^{-1}t_{\mathcal{B}} + (\nabla g_{\mathcal{B}}(z) + F_{\mathcal{B}})^T\widehat{\Delta z}\right].$$

By a direct comparison of these two expressions, and using $(D_{\mathcal{B}} + G_{\mathcal{B}})^{-1} = O(\mu)$, we find that

$$\widehat{\Delta\lambda}_{\mathcal{B}} - y = \delta_{\mathbf{u}}. \tag{5.8}$$

By combining (5.8) with (5.6b) and (5.6c), we find that

$$U^T(\Delta\lambda_{\mathcal{B}} - \widehat{\Delta\lambda}_{\mathcal{B}}) = \delta_{\mathbf{u}}, \qquad V^T(\Delta\lambda_{\mathcal{B}} - \widehat{\Delta\lambda}_{\mathcal{B}}) = \delta_{\mathbf{u}}/\mu. \tag{5.9}$$

For the "nonbasic" part $\widehat{\Delta\lambda}_{\mathcal{N}}$, we have from (5.2), (5.3b), (4.31), (5.6a), (5.7b), (4.2), and (3.16b) of Lemma 3.2 that

$$g_{\mathcal{N}}(z) = O(1), \qquad \Lambda_{\mathcal{N}}^{-1} = O(\mu^{-1}), \qquad \widehat{\Delta z} = O(\mu), \tag{5.10a}$$

$$(D_{\mathcal{N}} + G_{\mathcal{N}})^{-1} = (I + D_{\mathcal{N}}^{-1} G_{\mathcal{N}})^{-1} D_{\mathcal{N}}^{-1} = D_{\mathcal{N}}^{-1} + \mu\delta_{\mathbf{u}} = O(\mu). \tag{5.10b}$$

By using $t_{\mathcal{N}} = O(\mu^2)$ and applying the model (2.10) to the appropriate step in the procedure **condensed**, we obtain

$$\widehat{\Delta\lambda}_{\mathcal{N}} = (D_{\mathcal{N}} + G_{\mathcal{N}})^{-1}\left[g_{\mathcal{N}}(z) + f_{\mathcal{N}} - \Lambda_{\mathcal{N}}^{-1} t_{\mathcal{N}} + (\nabla g_{\mathcal{N}}(z) + F_{\mathcal{N}})^T \widehat{\Delta z} + \delta_{\mathbf{u}}\right] + \mu\delta_{\mathbf{u}}.$$

By comparing this expression with the corresponding exact formula, which is

$$\Delta\lambda_{\mathcal{N}} = D_{\mathcal{N}}^{-1}\left[g_{\mathcal{N}}(z) - \Lambda_{\mathcal{N}}^{-1} t_{\mathcal{N}} + \nabla g_{\mathcal{N}}(z)^T \Delta z\right],$$

and by using the bounds (5.10) and the fact that $f_{\mathcal{N}}$ and $F_{\mathcal{N}}$ have size $\delta_{\mathbf{u}}$, we obtain

$$\begin{aligned}
\widehat{\Delta\lambda}_{\mathcal{N}} - \Delta\lambda_{\mathcal{N}} &= \mu\delta_{\mathbf{u}} + (D_{\mathcal{N}} + G_{\mathcal{N}})^{-1}[f_{\mathcal{N}} + \delta_{\mathbf{u}}] + \\
&\qquad [(D_{\mathcal{N}} + G_{\mathcal{N}})^{-1} - D_{\mathcal{N}}^{-1}][g_{\mathcal{N}}(z) - \Lambda_{\mathcal{N}}^{-1} t_{\mathcal{N}}] + \\
&\qquad (D_{\mathcal{N}} + G_{\mathcal{N}})^{-1}(\nabla g_{\mathcal{N}}(z) + F_{\mathcal{N}})^T \widehat{\Delta z} - D_{\mathcal{N}}^{-1} \nabla g_{\mathcal{N}}(z)^T \Delta z \\
&= \mu\delta_{\mathbf{u}} + (D_{\mathcal{N}} + G_{\mathcal{N}})^{-1}[\nabla g_{\mathcal{N}}(z)^T(\widehat{\Delta z} - \Delta z) + F_{\mathcal{N}}^T \widehat{\Delta z}] \\
&\qquad [(D_{\mathcal{N}} + G_{\mathcal{N}})^{-1} - D_{\mathcal{N}}^{-1}] \nabla g_{\mathcal{N}}(z)^T \Delta z \\
&= \mu\delta_{\mathbf{u}}.
\end{aligned} \tag{5.11}$$

Finally, for the recovered step $\widehat{\Delta s}$, we have from the last step of procedure **condensed**, together with (3.11b), (5.2d), (5.7b), and (2.10) that

$$\widehat{\Delta s} = -(g(z) + f + s) - (\nabla g(z) + F)^T \widehat{\Delta z} + \mu\delta_{\mathbf{u}},$$

where the final term accounts for the rounding error (2.10) that arises from accumulating the terms in the sum, which are all of size $\delta_{\mathbf{u}}$ or $O(\mu)$. By substituting the expression for the exact Δs together with the estimates (5.2d) and (5.7b) on the sizes of the perturbation terms, we obtain

$$\begin{aligned}
\widehat{\Delta s} &= -(g(z) + s) - \nabla g(z)^T \Delta z - f - \nabla g(z)^T(\widehat{\Delta z} - \Delta z) - F^T \widehat{\Delta z} + \mu\delta_{\mathbf{u}} \\
&= \Delta s + \delta_{\mathbf{u}}.
\end{aligned} \tag{5.12}$$

We summarize the results obtained so far in the following theorem.

THEOREM 5.1. *Suppose that Assumption 4.1 holds. Then when the step $(\widehat{\Delta z}, \widehat{\Delta\lambda}, \widehat{\Delta s})$ is calculated in a finite-precision environment by using the procedure*

condensed *(and where, in particular, a backward stable method is used to solve the linear system for the* $\widehat{\Delta z}$ *component), we have that*

$$(\Delta z - \widehat{\Delta z}, U^T(\Delta\lambda_{\mathcal{B}} - \widehat{\Delta\lambda}_{\mathcal{B}}), \Delta s - \widehat{\Delta s}) = \delta_{\mathbf{u}}, \tag{5.13a}$$

$$V^T(\Delta\lambda_{\mathcal{B}} - \widehat{\Delta\lambda}_{\mathcal{B}}) = \delta_{\mathbf{u}}/\mu, \tag{5.13b}$$

$$\Delta\lambda_{\mathcal{N}} - \widehat{\Delta\lambda}_{\mathcal{N}} = \mu\delta_{\mathbf{u}}. \tag{5.13c}$$

This theorem extends the result of M. H. Wright [24] for accuracy of the computed solution of the condensed system by relaxing the LICQ assumption to MFCQ. When LICQ holds, the matrix V is vacuous, so the absolute error in all components is of size at most $\delta_{\mathbf{u}}$. The higher accuracy (5.13c) of the components $\widehat{\Delta\lambda}_{\mathcal{N}}$ (also noted in [24]) does not contribute significantly to the progress that can be made along the inexact direction $(\widehat{\Delta z}, \widehat{\Delta\lambda}, \widehat{\Delta s})$, in the sense of Section 5.3.

5.2. Termination of the Cholesky Algorithm. In deriving the estimate (5.6), we have assumed that a backward stable algorithm is used to solve (5.1). Because of (2.6), (2.7), and the SC condition, and the estimates of the sizes of the diagonals of D (from (4.2) and Lemma 3.2), it is easy to show that the matrix in (5.1) is positive definite for all sufficiently small μ. The Cholesky algorithm is therefore an obvious candidate for solving this system. However, the condition number of the matrix in (5.1) usually approaches ∞ as $\mu \downarrow 0$, raising the possibility that the Cholesky algorithm may break down when μ is small. A simple argument, which we now sketch, suffices to show that successful completion of the Cholesky algorithm can be expected under the assumptions we have used in our analysis so far.

We state first the following technical result. Since it is similar to one proved by Hager [16, Lemma 3] and Debreu [6, Theorem 3], its proof is omitted.

LEMMA 5.2. *Suppose that M and A are two matrices with the properties that M is symmetric and*

$$A^Tx = 0 \quad \Rightarrow \quad x^TMx \geq \alpha\|M\|\|x\|^2,$$

for some constant $\alpha > 0$. Then for all μ such that

$$0 < \mu < \bar{\mu} \overset{\text{def}}{=} \min\left(\frac{\alpha\|A\|^2}{4\|M\|}, \frac{\|A\|}{\alpha\|M\|}\right),$$

we have that

$$x^T(M + \mu^{-1}AA^T)x \geq \frac{\alpha}{2}\|x\|^2, \qquad \text{for all } x.$$

We apply this result to (5.1) by setting

$$\begin{aligned} M &= \mathcal{L}_{zz}(z,\lambda) + \nabla g_{\mathcal{N}}(z)D_{\mathcal{N}}^{-1}\nabla g_{\mathcal{N}}(z)^T = \mathcal{L}_{zz}(z,\lambda) + O(\mu), \\ A &= \mu^{1/2}\nabla g_{\mathcal{B}}(z)D_{\mathcal{B}}^{-1/2} \end{aligned}$$

(where again we use (4.2) and Lemma 3.2 to derive the order estimates). The conditions (2.6), (2.7), and strict complementarity ensure that this choice of M and A satisfies the assumptions of Lemma 5.2. The result then implies that the smallest singular value of the matrix in (5.1) is positive and of size $\Theta(1)$ for all values of μ

below a threshold that is also of size $\Theta(1)$. Since $D = O(\mu^{-1})$, the largest eigenvalue of this matrix is of size $O(\mu^{-1})$, so we have

$$\operatorname{cond}(\mathcal{L}_{zz}(z,\lambda) + \nabla g(z) D^{-1} \nabla g(z)^T) = O(\mu^{-1}). \tag{5.14}$$

(An estimate similar to this is derived by M. H. Wright [24, Theorem 3.2], under the LICQ assumption.) It is known from a result of Wilkinson (cited by Golub and Van Loan [12, p. 147]) that the Cholesky algorithm runs to completion if $q_n \delta_{\mathbf{u}} \operatorname{cond}(\cdot) \le 1$, where q_n is a modest quantity that depends polynomially on the dimension n of the matrix. By combining this result with (5.14), we conclude that for the matrix in (5.1), we can expect completion of the Cholesky algorithm whenever $\mu \gg \delta_{\mathbf{u}}$. That is, no new assumptions need to be added to those made in deriving the results of earlier sections.

We note that this situation differs a little from the case of linear programming where, because second-order conditions are not applicable, it is usually necessary to modify the Cholesky procedure to ensure that it runs to completion (see [31]).

5.3. Local Convergence with Computed Steps. We begin this section by showing how the quantities r_f, r_g, and μ change along the computed step $(\widehat{\Delta z}, \widehat{\Delta\lambda}, \widehat{\Delta s})$ obtained from the finite-precision implementation of the procedure **condensed**. We compare these with the changes that can be expected along the exact direction $(\Delta z, \Delta\lambda, \Delta s)$. We then consider the effects of these perturbations on an algorithm of the type in which the iterates are expected to satisfy the conditions (3.11). Rapidly convergent variants of these algorithms for linear programming problems usually allow the values of C and γ in these conditions to be relaxed, so that a near-unit step can be taken. We address the following question: If similar relaxations are allowed in an algorithm for nonlinear programming, are near-unit steps still possible when the steps contain perturbations of the type considered above?

We show in particular that for the computed search direction, the maximum step length that can be taken without violating the nonnegativity conditions on λ and s satisfies

$$1 - \hat{\alpha}_{\max} = \delta_{\mathbf{u}}/\mu + O(\mu), \tag{5.15}$$

while the reductions in r_f, r_g, and μ satisfy

$$\mu(\lambda + \alpha\widehat{\Delta\lambda}, s + \alpha\widehat{\Delta s}) = (1-\alpha)\mu + \delta_{\mathbf{u}} + O(\mu^2), \tag{5.16a}$$

$$r_f(z + \alpha\widehat{\Delta z}, \lambda + \alpha\widehat{\Delta\lambda}) = (1-\alpha) r_f + \delta_{\mathbf{u}} + O(\mu^2), \tag{5.16b}$$

$$r_g(z + \alpha\widehat{\Delta z}, s + \alpha\widehat{\Delta s}) = (1-\alpha) r_g + \delta_{\mathbf{u}} + O(\mu^2). \tag{5.16c}$$

The corresponding maximum steplength for the *exact* direction satisfies

$$1 - \alpha_{\max} = O(\mu), \tag{5.17}$$

while the reductions in r_f, r_g, and μ satisfy

$$\mu(\lambda + \alpha\Delta\lambda, s + \alpha\Delta s) = (1-\alpha)\mu + O(\mu^2), \tag{5.18a}$$

$$r_f(z + \alpha\Delta z, \lambda + \alpha\Delta\lambda) = (1-\alpha) r_f + O(\mu^2), \tag{5.18b}$$

$$r_g(z + \alpha\Delta z, s + \alpha\Delta s) = (1-\alpha) r_g + O(\mu^2). \tag{5.18c}$$

Our proof of the estimates (5.15) and (5.16) is tedious but not completely straightforward, and we have included it in the Appendix.

It is clear from (5.15) and (5.16) that the direction $(\widehat{\Delta z}, \widehat{\Delta \lambda}, \widehat{\Delta s})$ makes good progress toward the solution set $\mathcal{S}$. If the actual steplength α is close to its maximum value $\hat{\alpha}_{\max}$, in the sense that

$$\hat{\alpha}_{\max} - \alpha = \delta_{\mathbf{u}}/\mu + O(\mu), \tag{5.19}$$

we have by direct substitution in (5.15) and (5.16) that

$$\begin{aligned} \mu(\lambda + \alpha\widehat{\Delta \lambda}, s + \alpha\widehat{\Delta s}) &= \delta_{\mathbf{u}} + O(\mu^2), \\ r_f(z + \alpha\widehat{\Delta z}, \lambda + \alpha\widehat{\Delta \lambda}) &= \delta_{\mathbf{u}} + O(\mu^2), \\ r_g(z + \alpha\widehat{\Delta z}, s + \alpha\widehat{\Delta s}) &= \delta_{\mathbf{u}} + O(\mu^2). \end{aligned}$$

These formulae suggest that finite precision does not have an observable effect on the quadratic convergence rate of the underlying algorithm until μ drops below about $\sqrt{\mathbf{u}}$. Stopping criteria for interior-point methods usually include a condition such as $\mu \leq 10^4\mathbf{u}$ or $\mu \leq \sqrt{\mathbf{u}}$ (see, for example, [5]), so that μ is not allowed to become so small that the assumption $\mu \gg \mathbf{u}$ made in (4.7) is violated.

In making this back-of-the-envelope assessment, however, we have not taken into account the approximate centrality conditions (3.11), which must continue to hold (possibly in a relaxed form) at the new iterate. These conditions play a central role both in the analysis above and in the convergence analysis of the underlying "exact" algorithms, and also appear to be important in practice. Typically (see, for example, Ralph and Wright [23]), the conditions (3.11) are relaxed by allowing a modest increase in C and a modest decrease in γ on the rapidly convergent steps. We show in the next result that enforcement of these relaxed conditions is not inconsistent with taking a step length α that is close to $\hat{\alpha}_{\max}$, so that rapid convergence can still be observed even in the presence of finite-precision effects.

THEOREM 5.3. *Suppose that Assumption 4.1 holds. Then when the step* $(\widehat{\Delta z}, \widehat{\Delta \lambda}, \widehat{\Delta s})$ *is calculated in a finite-precision environment by using the procedure* **condensed**, *there is a constant* $\hat{C}$ *such that for all* $\tau \in [0, 1/2]$ *and all* α *satisfying*

$$\alpha \in [0, 1 - \hat{C}\tau^{-1}(\mathbf{u}/\mu + \mu)], \tag{5.20}$$

the following relaxed form of the approximate centrality conditions holds:

$$r_f(z + \alpha\widehat{\Delta z}, \lambda + \alpha\widehat{\Delta \lambda}) \leq C(1+\tau)\mu(\lambda + \alpha\widehat{\Delta \lambda}, s + \alpha\widehat{\Delta s}), \tag{5.21a}$$

$$r_g(z + \alpha\widehat{\Delta z}, s + \alpha\widehat{\Delta s}) \leq C(1+\tau)\mu(\lambda + \alpha\widehat{\Delta \lambda}, s + \alpha\widehat{\Delta s}), \tag{5.21b}$$

$$(\lambda_i + \alpha\widehat{\Delta \lambda}_i)(s_i + \alpha\widehat{\Delta s}_i) \geq \gamma(1-\tau)\mu(\lambda + \alpha\widehat{\Delta \lambda}, s + \alpha\widehat{\Delta s}), \quad \text{for all } i = 1, 2, \ldots, m, \tag{5.21c}$$

where C *is the constant from conditions (3.11). Moreover, when we set* α *to its upper bound in (5.20), we find that*

$$\delta(z + \alpha\widehat{\Delta z}, \lambda + \alpha\widehat{\Delta \lambda}) \leq \tau^{-1}(\delta_{\mathbf{u}} + O(\mu^2)). \tag{5.22}$$

Proof. From (3.11) and (5.16), we have that

$$\|r_f(z + \alpha\widehat{\Delta z}, \lambda + \alpha\widehat{\Delta \lambda})\|$$

$$\begin{aligned}
&= (1-\alpha)\|r_f\| + \delta_{\mathbf{u}} + O(\mu^2) \\
&\le C(1-\alpha)\mu + \delta_{\mathbf{u}} + O(\mu^2) \\
&= C(1+\tau)(1-\alpha)\mu - C\tau(1-\alpha)\mu + \delta_{\mathbf{u}} + O(\mu^2) \\
&= C(1+\tau)\mu(\lambda + \alpha\widehat{\Delta\lambda}, s + \alpha\widehat{\Delta s}) - C\tau(1-\alpha)\mu + \delta_{\mathbf{u}} + O(\mu^2).
\end{aligned}$$

We deduce that the required condition (5.21a) will hold provided that

$$\delta_{\mathbf{u}} + O(\mu^2) \le C\tau(1-\alpha)\mu.$$

Since by definition we have that $\delta_{\mathbf{u}} + O(\mu^2) \le \bar{C}(\mathbf{u} + \mu^2)$ for some positive constant $\bar{C}$, we find that a sufficient condition for the required inequality is that

$$(1-\alpha) \ge (\bar{C}/C)\tau^{-1}(\mathbf{u}/\mu + \mu),$$

which is equivalent to (5.20) for an obvious definition of $\hat{C}$. Identical logic can be applied to $\|r_g\|$ to yield a similar condition on α.

For the condition (5.21c), we have from (3.11) and (5.16) that

$$\begin{aligned}
&(\lambda_i + \alpha\widehat{\Delta\lambda}_i)(s_i + \alpha\widehat{\Delta s}_i) \\
&= (1-\alpha)\lambda_i s_i + \delta_{\mathbf{u}} + O(\mu^2) \\
&\ge (1-\alpha)\gamma\mu + \delta_{\mathbf{u}} + O(\mu^2) \\
&\ge \gamma\mu(\lambda + \alpha\widehat{\Delta\lambda}, s + \alpha\widehat{\Delta s}) + \delta_{\mathbf{u}} + O(\mu^2) \\
&\ge \gamma(1-\tau)\mu(\lambda + \alpha\widehat{\Delta\lambda}, s + \alpha\widehat{\Delta s}) + \gamma\tau(1-\alpha)\mu + \delta_{\mathbf{u}} + O(\mu^2).
\end{aligned}$$

Hence, the condition (5.21c) holds provided that

$$\gamma\tau(1-\alpha)\mu + \delta_{\mathbf{u}} + O(\mu^2) \ge 0.$$

Similar logic can be applied to this inequality to derive a bound of the type (5.20), after a possible adjustment of $\hat{C}$.

Finally, we obtain (5.22) by substituting $\alpha = 1 - \hat{C}\tau^{-1}(\mathbf{u}/\mu + \mu)$ into (5.16) and applying Theorem 3.3. (Note that, despite the relaxation of the centrality conditions (5.21), the result of Theorem 3.3 still holds; we simply modify the proof to replace C by $(3/2)C$ in (3.11a) and (3.11b), and γ by $\gamma/2$ in (3.11c).) □

6. The Augmented System. In this section, we consider the case in which the augmented system (3.9) (equivalently, (4.1)) is solved to obtain $(\Delta z, \Delta\lambda)$, while the remaining step component Δs is recovered from (3.8). The formal specification for this procedure is as follows:

```
procedure augmented
given the current iterate $(z, \lambda, s)$
    form the coefficient matrix and right-hand side for (4.1);
    solve (4.1) to obtain $(\Delta z, \Delta\lambda)$;
    set $\Delta s = -(g(z) + s) - \nabla g(z)^T \Delta z$.
```

Much of our work in analyzing the augmented system form (4.1) has already been performed in Section 4; the main error result is simply Corollary 4.4. However, we can apply this result only if the floating-point errors made in evaluating and solving this system satisfy the assumptions of this corollary. In particular, we need to show that the perturbation matrices E_{ij}, $i, j = 1, 2, 3$ in (4.34) satisfy the estimates (4.28).

This task is not completely straightforward. Unlike the condensed and full-system cases, it is not simply a matter of assuming that a backward-stable algorithm has been used to solve the system (4.1). The reason is that the largest terms in the coefficient matrix in (3.9)—the diagonal elements in the matrix $D_{\mathcal{N}}$—have size $O(\mu^{-1})$. The usual analysis of backward-stable algorithms represents the floating-point errors as a perturbation of the entire coefficient matrix whose size is bounded by $\delta_{\mathbf{u}}$ times the matrix norm—in this case, $\delta_{\mathbf{u}}/\mu$. However, Corollary 4.4 requires some elements of the perturbation matrix to be smaller than this estimate; in particular, the submatrices E_{12}, E_{21}, and E_{22} need to be of size $\delta_{\mathbf{u}}$. Therefore, we need to look closely at the particular algorithms used to solve (4.1) to see whether they satisfy the following condition.

CONDITION 6.1. *The solution obtained by applying the algorithm in question to the system (4.1) in floating-point arithmetic is the exact solution of a perturbed system in which the perturbations of the coefficient matrix satisfy the estimates (4.28), while the right-hand side is unperturbed.*

We focus on diagonal pivoting methods, which take a symmetric matrix T and produce a factorization of the form

$$PTP^T = LYL^T, \tag{6.1}$$

where P is a permutation matrix, L is unit lower triangular, and Y is block diagonal, with a combination of 1×1 and symmetric 2×2 blocks. The best-known methods of this class are due to Bunch and Parlett [3] and Bunch and Kaufman [2], while Duff et al. [7] and Fourer and Mehrotra [10] have described sparse variants. These algorithms differ in their selection criteria for the 1×1 and 2×2 pivot blocks. In our case, the presence of the diagonal elements of size $\Theta(\mu^{-1})$ (from the submatrix $D_{\mathcal{N}} = \Lambda_{\mathcal{N}}^{-1} S_{\mathcal{N}}$) and their place in these pivot blocks are crucial to the result.

We start by stating a general result of Higham [18] concerning backward stability that applies to all diagonal pivoting schemes. We then examine the Bunch-Kaufman scheme, showing that the large diagonals appear only as 1×1 pivots and that this algorithm satisfies Condition 6.1. (In [18, Theorem 4.2], Higham actually proves that the Bunch-Kaufman scheme is backward stable in the normwise sense, but this result is not applicable to our context, for the reasons mentioned above.)

Next, we briefly examine the Bunch-Parlett method, showing that it starts out by selecting all the large diagonal elements in turn as 1×1 pivots, before going on to factor the remaining matrix, whose elements are all $O(1)$ in size. This method also satisfies Condition 6.1. We then examine the sparse diagonal pivoting approaches of Duff et al. [7] and Fourer and Mehrotra [10], which may not satisfy Condition 6.1, because of the possible presence of 2×2 pivots in which one of the diagonals has size $\Theta(\mu^{-1})$. These algorithms can be modified in simple ways to overcome this difficulty, possibly at the expense of higher density in the L factor. We then mention Gaussian elimination with pivoting and refer to previous results in the literature to show that this approach satisfies Condition 6.1. Finally, we state a result like Theorem 5.3 about convergence of a finite-precision implementation of an algorithm based on the augmented system form.

Higham [18, Theorem 4.1] proves the following result.

THEOREM 6.1. *Let T be an $\bar{n} \times \bar{n}$ symmetric matrix, and let $\hat{x}$ be the computed solution to the linear system $Tx = b$ produced by a method that yields a factorization of the form (6.1), with any diagonal pivoting strategy. Assume that, during recovery of the solution, the subsystems that involve the 2×2 diagonal blocks are solved via*

Gaussian elimination with partial pivoting. Then we have that

$$(T+\Delta T)\hat{x} = b, \qquad |\Delta T| \le \delta_{\mathbf{u}}(|T| + P^T|\hat{L}||\hat{Y}||\hat{L}^T|P) + \delta_{\mathbf{u}}^2, \tag{6.2}$$

where $\hat{L}$ *and* $\hat{Y}$ *are the computed factors, and* $|A|$ *denotes the matrix formed from* A *by replacing all its elements by their absolute values.*

In Higham's result, the coefficient of $\mathbf{u}$ in the $\delta_{\mathbf{u}}$ term is actually a linear polynomial in the dimension of the system. The partial pivoting strategy for the 2×2 systems can actually be replaced by any method for which the computed solution of $Ry = d$ satisfies $(R + \Delta R)\hat{y} = d$, where R is the 2×2 matrix in question and $|\Delta R| \le \delta_{\mathbf{u}}|R|$. This property was also key in an earlier paper of S. Wright [28], who derived a result similar to Theorem 6.1 in the context of the augmented systems that arise from interior-point methods for linear programming.

All the procedures below have the property that the growth in the maximum element size in the remaining submatrix is bounded by a modest quantity at each individual step of the factorization. (In the case of Bunch-Kaufman and Bunch-Parlett, this bound averages about 2.6 per elimination step; see Golub and Van Loan [12, Section 4.4.4].) As with Gaussian elimination with partial pivoting, exponential element growth is possible, so that L and Y in (6.1) contain much larger elements than the original matrix T. Such behavior is, however, quite rare and is confined to pathological cases and certain special problem classes. In our analysis below, we make the safe assumption that catastrophic growth of this kind does not occur.

6.1. The Bunch-Kaufman Procedure.

At each iteration, the Bunch-Kaufman procedure chooses either a 1×1 or 2×2 pivot by examining at most two columns of the remaining matrix, that is, the part of the matrix that remains to be factored at this stage of the process. It makes use of quantities χ_i defined by

$$\chi_i = \max_{j \,|\, j \ne i} |T_{ij}|,$$

where in this case T denotes the remaining matrix. We define the pivot selection strategy for the first step of the factorization process. The entire algorithm is obtained by applying this procedure recursively to the remaining submatrix.

```
set $\nu = (1+\sqrt{17})/8$;
calculate $\chi_1$, and store the index $r$ for which $\chi_1 = |T_{r1}|$;
if $|T_{11}| \ge \nu\chi_1$
    choose $T_{11}$ as a $1 \times 1$ pivot;
else
    calculate $\chi_r$;
    if $\chi_r|T_{11}| \ge \nu\chi_1^2$
        choose $T_{11}$ as a $1 \times 1$ pivot;
    else if $|T_{rr}| \ge \nu\chi_r$
        choose $T_{rr}$ as a $1 \times 1$ pivot;
    else
        choose a $2 \times 2$ pivot with diagonals $T_{11}$ and $T_{rr}$;
    end if
end if.
```

For each choice of pivot, the permutation matrix P_1 is chosen so that the desired 1×1 or 2×2 pivot is in the upper left of the matrix $P_1 T P_1^T$. If one writes

$$P_1 T P_1^T = \begin{bmatrix} R & C^T \\ C & \hat{T} \end{bmatrix},$$

where R is the chosen pivot, the first step of the factorization yields

$$P_1 T P_1^T = \begin{bmatrix} I & \\ CR^{-1} & I \end{bmatrix} \begin{bmatrix} R & \\ & \bar{T} \end{bmatrix} \begin{bmatrix} I & R^{-1}C^T \\ & I \end{bmatrix}, \tag{6.3}$$

where $\bar{T} = \hat{T} - CR^{-1}C^T$ is the matrix remaining after this stage of the factorization.

At the first step of the factorization, the quantities χ_1 and χ_r (if calculated) both have size $O(1)$, since the large elements of this matrix occur only on the diagonal. Since a 2×2 pivot is chosen only if

$$|T_{11}| < \nu\chi_1 \quad \text{and} \quad |T_{rr}| < \nu\chi_r,$$

it follows immediately that both diagonals in a 2×2 pivot must be $O(1)$. Hence, the pivot chosen by this procedure is one of three types:

(6.4a) 1×1 pivot of size $O(1)$;

(6.4b) 2×2 pivot in which both diagonals have size $O(1)$;

(6.4c) 1×1 pivot of size $\Theta(\mu^{-1})$.

In fact, the pivots are one of the types (6.4) at *all* stages of the factorization, not just the first stage. The reason is that the updated matrix $\bar{T}$ in (6.3) has the same essential form as the original matrix T—its elements are all of size $O(1)$ except for some large diagonal elements of size $\Theta(\mu^{-1})$. We demonstrate this claim by showing that the update $CR^{-1}C$ that is applied to the remaining matrix in (6.3) is a matrix whose elements are of size at most $O(1)$, regardless of the type of pivot, so that it does not disturb the essential structure of the remaining matrix. When the pivots are of type (6.4a) and (6.4b), the standard argument of Bunch and Kaufman [2] can be applied to show that the norm of $CR^{-1}C^T$ is at most a modest multiple of $\|C\|$. We know that $\|C\| = O(1)$, since C consists only of off-diagonal elements, so we conclude that $\|CR^{-1}C^T\| = O(1)$ in this case as claimed. For the other pivot type (6.4c), we have $R = \Theta(\mu^{-1})$ and $C = O(1)$, so the elements of $CR^{-1}C^T$ have size $O(\mu)$, and the claim holds in this case too.

In the rest of this subsection, we show by using Theorem 6.1 that Condition 6.1 holds for the Bunch-Kaufman algorithm. In fact, we prove a stronger result: When T in Theorem 6.1 is the matrix (4.1), the perturbation matrix ΔT contains elements of size $\delta_{\mathbf{u}}$, except in those diagonal locations corresponding to the elements of $D_{\mathcal{N}}$, where they may be as large as $\delta_{\mathbf{u}}/\mu$. Given the bound on $|\Delta T|$ in (6.2), we need only to show that $P^T|\hat{L}||\hat{Y}||\hat{L}|^T P$ has the desired structure. In fact, it suffices to show that the exact factor product $P^T|L||Y||L|^T P$ has the structure in question, since the difference between these two products is just $\delta_{\mathbf{u}}$ in size.

We demonstrate this claim inductively, using a refined version of the arguments from Higham [18, Section 4.3] for some key points, and omitting some details. For simplicity, and without loss of generality, we assume that $P = I$.

When $\bar{n} = 1$ (that is, T is 1×1), we have that $L = 1$ and $Y = T$, and the result holds trivially. When $\bar{n} = 2$, there are three cases to consider. If the matrix contains no elements of size $\Theta(\mu^{-1})$, then the analysis for general matrices can be used to show that $|L||Y||L|^T = O(1)$, as required. If either or both diagonal elements have size $\Theta(\mu^{-1})$, then both pivots are 1×1, and the factors have the form

$$L = \begin{bmatrix} 1 & 0 \\ T_{21}/T_{11} & 1 \end{bmatrix}, \qquad Y = \begin{bmatrix} T_{11} & 0 \\ 0 & T_{22} - T_{21}^2/T_{11} \end{bmatrix}. \tag{6.5}$$

Two cases arise.

(i) A diagonal of size $O(1)$ is accepted as the first pivot and moved (if necessary) to the $(1,1)$ position. We then have

$$|T_{11}| \geq \nu\chi_1 = \nu\chi_2 = \nu|T_{21}|,$$

and therefore $|T_{21}/T_{11}| \leq 1/\nu$ and hence $|T_{21}^2/T_{11}| \leq |T_{21}|/\nu = O(1)$. If the $(2,2)$ diagonal is also $O(1)$, we have that $L = O(1)$ and $Y = O(1)$, and we are done. Otherwise, $T_{22} = \Theta(\mu^{-1})$, and so the $(2,2)$ element of Y satisfies this same estimate. We conclude from (6.5) that $|L||Y||L|^T$ also has an $\Theta(\mu^{-1})$ element in the $(2,2)$ position and $O(1)$ elements elsewhere.

(ii) A diagonal of size $\Theta(\mu^{-1})$ is accepted as the first pivot and moved (if necessary) to the $(1,1)$ position. We then have

$$|T_{21}/T_{11}| = O(\mu), \qquad |T_{21}^2/T_{11}| = O(\mu).$$

It follows from (6.5) that

$$|L||Y||L|^T = \begin{bmatrix} |T_{11}| & |T_{21}| \\ |T_{21}| & |T_{22}| + O(\mu) \end{bmatrix},$$

which obviously has the desired structure.

We now assume that our claim holds for some dimension $\bar{n} \geq 2$, and we prove that it continues to hold for dimension $\bar{n}+1$. Using the notation of (6.3) (assuming that $P_1 = I$), and denoting the factorization of the Schur complement $\bar{T}$ in (6.3) by $\bar{T} = \bar{L}\bar{Y}\bar{L}^T$, we have that

$$T = LYL^T = \begin{bmatrix} I & \\ CR^{-1} & \bar{L} \end{bmatrix} \begin{bmatrix} R & \\ & \bar{Y} \end{bmatrix} \begin{bmatrix} I & R^{-1}C^T \\ & \bar{L}^T \end{bmatrix}. \tag{6.6}$$

It follows that

$$|L||Y||L|^T = \begin{bmatrix} |R| & |R||R^{-1}C^T| \\ |CR^{-1}||R| & |CR^{-1}||R||R^{-1}C^T| + |\bar{L}||\bar{Y}||\bar{L}|^T \end{bmatrix}. \tag{6.7}$$

Since, as we mentioned above, the norm of $CR^{-1}C$ is at most $O(1)$, the Schur complement $\bar{T} = \hat{T} - CR^{-1}C^T$ has size $O(1)$ except for large $\Theta(\mu^{-1})$ elements in the same locations as in the original matrix. Hence, by our inductive hypothesis, $|\bar{L}||\bar{Y}||\bar{L}|^T$ has a similar structure, and we need to show only that the effects of the first step of the factorization (6.3) do not disturb the desired structure.

For the case in which R is a pivot of type either (6.4a) and (6.4b), Higham [18, Section 4.3] shows all elements of both $|CR^{-1}||R|$ and $|CR^{-1}||R||R^{-1}C^T|$ are bounded by a modest multiple of either χ_1 (if T_{11} was selected as the pivot because it passed the test $|T_{11}| \geq \nu\chi_1$) or $(\chi_1 + \chi_r)$, where r is the "other" column considered during the selection process. In our case, this observation implies that both $|CR^{-1}||R|$ and $|CR^{-1}||R||R^{-1}C^T|$ have size $O(1)$. By combining these observations with those of the preceding paragraph, we conclude that for pivots of types (6.4a) and (6.4b), "large" elements of the matrix in (6.7) occur only in the diagonal locations originally occupied by $D_{\mathcal{N}}$.

For the remaining case—pivots of type (6.4c)—we have that C has size $O(1)$ while R^{-1} has size $O(\mu)$. Therefore, $|CR^{-1}||R|$ has size $O(1)$ and $|CR^{-1}||R||R^{-1}C^T|$ has size $O(\mu)$, while $|R|$, which occupies the $(1,1)$ position in the matrix (6.7), just as it did in the original matrix T, has size $\Theta(\mu^{-1})$. We conclude that the desired structure holds in this case as well.

We conclude from this discussion that Condition 6.1 holds for the Bunch-Kaufman procedure. We show later that the perturbations arising from other sources, namely, roundoff and cancellation in the evaluation of the matrix and right-hand side, also satisfy the conditions of Corollary 4.4, so this result can be used to bound the error in the computed steps.

Finally, we note that it is quite possible for pivots of types (6.4a) and (6.4b) to be chosen while diagonal elements of size $\Theta(\mu^{-1})$ still remain in the submatrix. Therefore, a key assumption of the analysis of Forsgren, Gill, and Shinnerl [9, Theorem 4.4]—namely, that all the diagonals of size $\Theta(\mu^{-1})$ are chosen as 1×1 pivots before any of the other diagonals are chosen—may not be satisfied by the Bunch-Kaufman procedure.

6.2. The Bunch-Parlett Procedure. The Bunch-Parlett procedure is conceptually simpler but more expensive to implement than Bunch-Kaufman, since it requires $O(n^2)$ (rather than $O(n)$) comparisons at each step of the factorization. The pivot selection strategy is as follows.

```
set $\nu = (1 + \sqrt{17})/8$;
calculate $\chi_{\text{off}} = |T_{rs}| = \max_{i \neq j} |T_{ij}|$, $\chi_{\text{diag}} = |T_{pp}| = \max_i |T_{ii}|$;
if $\chi_{\text{diag}} \geq \nu \chi_{\text{off}}$
    choose $T_{pp}$ as the $1 \times 1$ pivot;
else
    choose the $2 \times 2$ pivot whose off-diagonal element is $T_{rs}$;
end if.
```

The elimination procedure then follows as in (6.3).

It is easy to show that the Bunch-Parlett procedure starts by selecting all the diagonals of size $\Theta(\mu^{-1})$ in turn as 1×1 pivots. (Because of this property, it satisfies the key assumption of [9] mentioned at the end of the preceding section.) The update $CR^{-1}C^T$ generated by each of these pivot steps has size only $O(\mu)$, so the matrix that remains after this phase of the factorization contains only $O(1)$ elements. The remaining pivots are then a combination of types (6.4a) and (6.4b).

By using the arguments of the preceding subsection in a slightly simplified form, we can show that Condition 6.1 holds for this procedure as well.

6.3. Sparse Diagonal Pivoting. For large instances of (1.1), the Bunch-Kaufman and Bunch-Parlett procedures are usually inefficient because they do not try to maintain sparsity in the lower triangular factor L. Sparse variants of these algorithms, such as those of Duff et al. [7] and Fourer and Mehrotra [10], use pivot selection strategies that combine stability considerations with Markowitz-like estimates of the amount of fill-in that a candidate pivot will cause in the remaining matrix.

At each stage of the factorization, both algorithms examine a roster of possible 1×1 and 2×2 pivots, starting with those that would create the least fill-in. As soon as a pivot is found that meets the stability criteria described below, it is accepted. Both algorithms prefer to use 1×1 pivots where possible.

For candidate 1×1 pivots, Duff et al. [7, p. 190] use the following stability criterion:

$$|R^{-1}| \|C\|_\infty \leq \rho, \tag{6.8}$$

where the notation R and C is from (6.3) and $\rho \in [2, \infty)$ is some user-selected parameter that represents the tolerable growth factor at each stage of the factorization.

For a 2×2 pivot, the criterion is

$$|R^{-1}| \begin{bmatrix} \|C_{\cdot,1}\|_\infty \\ \|C_{\cdot,2}\|_\infty \end{bmatrix} \le \begin{bmatrix} \rho \\ \rho \end{bmatrix}, \tag{6.9}$$

where $C_{\cdot,1}$ and $C_{\cdot,2}$ are the two columns of C. The stability criteria of Fourer and Mehrotra [10] are similar.

As they stand, the stability tests (6.8) and (6.9) do not necessarily restrict the choice of pivots to the three types (6.4). If a 1×1 pivot of size $\Theta(\mu^{-1})$ is ever considered for structural reasons, it will pass the test (6.8) (the left-hand side of this expression will have size $O(\mu)$) and therefore will be accepted as a pivot. However, it is possible that 2×2 pivots in which one or both diagonals have size $\Theta(\mu^{-1})$ may pass the test (6.9) and may therefore be accepted. Although the test (6.9) ensures that the size of the update $CR^{-1}C^T$ is modest (so that the update $\bar{T} = \hat{T} - CR^{-1}C^T$ does not disturb the large-diagonal structure of $\hat{T}$), there is no obvious assurance that the matrix $|L||Y||L|^T$ in (6.7) mirrors the structure of $|T|$, in terms of having the large diagonal elements in the same locations. The terms $|CR^{-1}||R|$ and $|CR^{-1}||R||R^{-1}C^T|$ in (6.7) may not have size $O(1)$, as they do for pivots of the three types (6.4) arising from the Bunch-Kaufman and Bunch-Parlett selection procedures.

The Fourer-Mehrotra algorithm does, however, rule out the possibility of a 2×2 pivot in which *both* diagonals are of size $\Theta(\mu^{-1})$. It considers a 2×2 candidate only if one of its diagonal elements has previously been considered as a 1×1 pivot but failed the stability test. However, if either of the diagonals had been subjected to the test (6.8), they would have been accepted, as noted in the preceding paragraph, so this situation cannot occur.

If the sparse algorithms are modified to ensure that all pivots have one of the three types (6.4), and all continue to satisfy the stability tests (6.8) or (6.9), then simple arguments (simpler than those of Section 6.1!) can be applied to show that Condition 6.1 is satisfied. One possible modification that achieves the desired efffect is to require that a 2×2 pivot be allowed only if *both* its diagonals have previously been considered as 1×1 pivots but failed the stability test (6.8).

6.4. Gaussian Elimination. Another possibility for solving the system (4.1) is to ignore its symmetry and apply a Gaussian elimination algorithm, with row and/or column pivoting to preserve sparsity and prevent excessive element growth. Such a strategy satisfies Condition 6.1. In [25], the author uses a result of Higham [17] to show that the effects of the large diagonal elements are essentially confined to the columns in which they appear. Assuming that the pivot sequence is chosen to prevent excessive element growth in the remaining matrix, and using the notation of (4.27) and (4.28), we can account for the effects of roundoff error in Gaussian elimination with perturbations in the coefficient matrix that satisfy the following estimates:

$$E_{11}, E_{21}, E_{31}, E_{12}, E_{22}, E_{32} = \delta_{\mathbf{u}}, \qquad E_{13}, E_{23}, E_{33} = \delta_{\mathbf{u}}/\mu.$$

These certainly satisfy the conditions (4.28), so Condition 6.1 holds.

6.5. Local Convergence with the Computed Steps. We can now state a formal result to show that when the evaluation errors are taken into account as well as the roundoff errors from the factorization/solve procedure discussed above, the accuracies of the computed steps obtained from the procedure **augmented**, implemented in finite precision, satisfy the same estimates as for the corresponding steps obtained from the procedure **condensed**. The result is analogous to Theorem 5.1.

Theorem 6.2. *Suppose that Assumption 4.1 holds. Then when the step* $(\widehat{\Delta z}, \widehat{\Delta\lambda}, \widehat{\Delta s})$ *is calculated in a finite-precision environment by using the procedure* **augmented**, *where the algorithm used to solve (4.1) satisfies Condition 6.1, we have*

$$(\Delta z - \widehat{\Delta z}, U^T(\Delta\lambda_{\mathcal{B}} - \widehat{\Delta\lambda}_{\mathcal{B}}), \Delta s - \widehat{\Delta s}) = \delta_{\mathbf{u}}, \tag{6.10a}$$

$$V^T(\Delta\lambda_{\mathcal{B}} - \widehat{\Delta\lambda}_{\mathcal{B}}) = \delta_{\mathbf{u}}/\mu, \tag{6.10b}$$

$$\Delta\lambda_{\mathcal{N}} - \widehat{\Delta\lambda}_{\mathcal{N}} = \delta_{\mathbf{u}}. \tag{6.10c}$$

Proof. The proof follows from Corollary 4.4 when we show that the perturbations to (4.1) from all sources—evaluation of the matrix and right-hand side as well as the factorization/solution procedure—satisfy the bounds required by this earlier result.

Because of Condition 6.1, perturbations arising from the factorization/solution procedure satisfy the bounds (4.28). The expressions (5.2) show that the errors arising from evaluation of $\mathcal{L}_{zz}(z,\lambda)$, $\mathcal{L}_z(z,\lambda)$, $\nabla g(z)$, and $g(z)$ are all of size $\delta_{\mathbf{u}}$, and hence they too satisfy the required bounds. Similarly to (5.3), evaluation of $D_{\mathcal{B}}$ and $D_{\mathcal{N}}$ yields errors of relative size $\delta_{\mathbf{u}}$, that is,

$$\text{computed } D_{\mathcal{B}} \leftarrow D_{\mathcal{B}} + G_{\mathcal{B}}, \quad G_{\mathcal{B}} = \mu\delta_{\mathbf{u}}, \tag{6.11a}$$

$$\text{computed } D_{\mathcal{N}} \leftarrow D_{\mathcal{N}} + G_{\mathcal{N}}, \quad G_{\mathcal{N}} = \delta_{\mathbf{u}}/\mu, \tag{6.11b}$$

where $G_{\mathcal{B}}$ and $G_{\mathcal{N}}$ are diagonal matrices.

We now obtain all the estimates in (6.10) by a direct application of Corollary 4.4, with the exception of the estimate for $(\Delta s - \widehat{\Delta s})$. Since the expressions for recovering Δs are identical in procedures **condensed** and **augmented**, we can apply expression (5.12) from Section 5.1 to deduce that the desired estimate holds for this component as well. ☐

The only difference between the error estimates of Theorem 5.1 for the condensed system and those obtained above for the augmented system is that the $\widehat{\Delta\lambda}_{\mathcal{N}}$ components are slightly less accurate in the augmented case. If we work through the analysis of Section 5.3 with the estimate (6.10c) replacing (5.13c), we find that the main results are unaffected. Therefore, we conclude this section by stating without proof a result similar to Theorem 5.3.

Theorem 6.3. *Suppose that all the assumptions of Theorem 5.3 hold, except that the step* $(\widehat{\Delta z}, \widehat{\Delta\lambda}, \widehat{\Delta s})$ *is calculated by using the procedure* **augmented** *with a factorization/solution algorithm that satisfies Condition 6.1. Then the conclusions of Theorem 5.3 hold.*

7. Numerical Illustration. We illustrate the results of Sections 5 and 6 using the two-variable example (2.8). Consider a simple algorithm that takes steps satisfying (3.8) with t set rather arbitrarily to $t = \mu^2 e$. (The search directions thus used are like those generated in the later stages of a practical primal-dual algorithm such as Mehrotra's algorithm [20].) We start this algorithm from the point

$$z_0 = (1/30, 1/9)^T, \quad \lambda_0 = (1, 1/5)^T, \quad s_0 = (1/10, 1/2).$$

(It is easy to check that the conditions (3.11) are satisfied at this point for a modest value of C.) At each step we calculated $\hat\alpha_{\max}$, defined in Section 5.3, and took an actual step of $.99\hat\alpha_{\max}$.

iter	$\log \mu$	$\log \lVert\widehat{\Delta z}\rVert$	$\log \lVert U^T \widehat{\Delta\lambda}_{\mathcal{B}}\rVert$	$\log \lVert V^T \widehat{\Delta\lambda}_{\mathcal{B}}\rVert$	$\hat{\alpha}_{\max}$	λ^T
0	-1.0	-0.9	-1.9	-1.9	.9227	(1.0,.2)
1	-2.7	-1.5	-1.3	-1.2	.9193	(.99,.19)
⋮						
5	-9.4	-6.7	-6.3	-4.6	1.0	(1.0,.23)
6	-11.4	-8.7	-8.3	-5.9	1.0	(1.0, .23)
7	-13.4	-10.7	-10.3	-3.8	.9999	(1.0,.23)
8	-15.4	-12.7	-12.3	-1.2	.9439	(1.0,.23)
9	-17.1	-13.9	-13.4	-0.6	.9723	(1.1,.20)

TABLE 7.1

Iteration sequence for PDIP method, with steps computed by solving the augmented system.

We programmed the method in Matlab, using the form (4.1) of the linear equations and using Gaussian elimination to solve the system (Section 6.4). From Theorem 6.2, the estimates (6.10) apply to this case.

Results are tabulated in Table 7.1. Note first the size of the component $\lVert V^T \widehat{\Delta\lambda}_{\mathcal{B}}\rVert$, which grows as μ decreases below $\mathbf{u}^{1/2}$, in accordance with (5.13b) and (6.10b). (We cannot tabulate the difference $\lVert V^T(\widehat{\Delta\lambda}_{\mathcal{B}} - \Delta\lambda_{\mathcal{B}}\rVert)$ because of course we do not know the true step $(\Delta z, \Delta\lambda, \Delta s)$, but since the true step has size $O(\mu)$ (Corollary 4.3), the error is dominated by the term $V^T \widehat{\Delta\lambda}_{\mathcal{B}}$ in any case.) As predicted by (5.15), the maximum step $\hat{\alpha}_{\max}$ becomes significantly smaller than 1 as μ is decreased below $\mathbf{u}^{1/2}$. As indicated by (5.16), however, good progress still can be made along this direction (in the sense of reducing μ and the norms of the residuals r_f and r_g) almost until μ reaches the level of $\mathbf{u}$. In fact, between iterations 5 and 8 we see the reduction factor of 100 that we would expect by moving a distance of .99 along a direction that is close to the pure Newton direction. The component with the large error—$V^T \widehat{\Delta\lambda}_{\mathcal{B}}$—does not interfere with the rapid convergence, but only causes the λ iterates to move tangentially to $\mathcal{S}_\lambda$. This effect may be noted in the final iterate where the value of λ changes significantly. In some cases, when the current λ is near the edge of the set $\mathcal{S}_\lambda$, this error may result in a severe curtailment of the step length. We do not tabulate the error in $\widehat{\Delta\lambda}_{\mathcal{N}}$ (see (5.13c), (6.10c)), which is too small to be observed.

8. Summary and Conclusions. In this paper, we have analyzed the finite-precision implementation of a primal-dual interior point method whose convergence rate is theoretically superlinear. We have made the standard assumptions that appear in most analyses of local convergence of nonlinear programming algorithms and path-following algorithms, with one significant exception: The assumption of linearly independent active constraint gradients is replaced by the weaker Mangasarian-Fromovitz constraint qualification, which is equivalent to boundedness of the set of optimal Lagrange multipliers. Because of this assumption, it is possible that all reasonable formulations of the step equations—the linear system that needs to be solved to obtain the search direction—are ill conditioned, so it is not obvious that the numerical errors that occur when this system is solved in finite precision do not eventually render the computed search direction useless. We show that although the error in the computed step may indeed become large as μ decreases, most of the error is restricted to a subspace that does not matter, namely, the null space of the Jacobian of the active constraints. Although this error causes the computed iterates to "slip" in a tangential direction to the optimal Lagrange multiplier set, it does not interfere with rapid convergence of the iterates to the primal-dual solution set.

We found that the centrality conditions (3.11), which are usually applied in path-following methods, played a crucial role in the analysis, since they enabled us to establish the estimates (3.16) in Lemma 3.2 concerning the sizes of the basic and nonbasic components of s and λ near the solution set. The analysis of Section 4, culminating in Corollary 4.4, finds bounds on the errors induced in step components by certain structured perturbations of the step equations. We show in the same section that the exact step is $O(\mu)$, allowing the local convergence analysis of Ralph and Wright [22] to be extended from convex programs to nonlinear programs.

In Sections 5 and 6 we apply the general results of Section 4 to the two most obvious ways of formulating and solving the step equations; namely, as a "condensed" system involving just the primal variables z, or as an "augmented" system involving both z and the Lagrange multipliers λ. In each case, the errors introduced in finite-precision implementation have the structure of the perturbations analyzed in Section 4, so the error bounds obtained in Corollary 4.4 apply. In Section 5.3 (whose analysis also applies to the computed solutions analyzed in Section 6), we show that the potentially large error component discussed above does not interfere appreciably with the near-linear decrease of the quantities μ, r_f, and r_g to zero along the computed steps, indicating that until μ becomes quite close to $\mathbf{u}$, the convergence behavior predicted by the analysis of the "exact" algorithm will be observed in the finite-precision implementation. We conclude in Section 7 with a numerical illustration of our major observervaions on a simple problem with two variables and two constraints, first introduced in Section 2.

Acknowledgments. Many thanks are due to an anonymous referee for a close and careful reading of the paper and for many helpful suggestions.

Appendix A.

Justification of the Estimates (5.15) and (5.16).

To prove (5.15), we use analysis similar to that of S. Wright [31]. From the definition (3.5) of μ, and the centrality condition (3.11c), we have that

$$\lambda_i s_i = \Theta(\mu), \quad \text{for all } i = 1, 2, \ldots, m.$$

Hence, from the third block row of (3.8) and the assumption (3.7) on the size of t, we have that

$$\text{(A.1)} \qquad \frac{\Delta\lambda_i}{\lambda_i} + \frac{\Delta s_i}{s_i} = -1 - \frac{t_i}{s_i\lambda_i} = -1 + O(\mu), \qquad \text{for all } i = 1, 2, \ldots, m.$$

We have from Lemma 3.2 and (4.31) that $\Delta\lambda_i/\lambda_i = O(\mu)$ for all $i \in \mathcal{B}$. Hence, by using (3.16a) from (3.2) together with (A.1), we obtain

$$\text{(A.2)} \qquad \Delta s_i = -s_i + O(\mu^2), \qquad \text{for all } i \in \mathcal{B}.$$

For the computed step components $\widehat{\Delta s}_{\mathcal{B}}$, we have by combining (5.13a) with (A.2) that

$$\text{(A.3)} \qquad \widehat{\Delta s}_i = -s_i + \delta_{\mathbf{u}} + O(\mu^2), \qquad \text{for all } i \in \mathcal{B}.$$

Therefore, if $s_i + \alpha\widehat{\Delta s}_i = 0$ for some $i \in \mathcal{B}$ and some $\alpha \in [0,1]$, we have by using (3.16a) again that

$$s_i + \alpha(-s_i + \delta_{\mathbf{u}} + O(\mu^2)) = 0$$

$$\begin{aligned} &\Rightarrow (1-\alpha)s_i = \delta_{\mathbf{u}} + O(\mu^2) \\ &\Rightarrow (1-\alpha) = \delta_{\mathbf{u}}/\mu + O(\mu), \qquad \text{for any } i \in \mathcal{B}. \end{aligned} \tag{A.4}$$

Meanwhile, for $i \in \mathcal{N}$, we have from Lemma 3.2, (4.31), and (5.13a) that

$$s_i + \alpha\widehat{\Delta s}_i > 0, \qquad \text{for all } i \in \mathcal{N} \text{ and all } \alpha \in [0,1], \tag{A.5}$$

so the components $\widehat{\Delta s}_{\mathcal{N}}$ do not place a limit on the step length bound $\hat{\alpha}_{\max}$. For the components $\widehat{\Delta\lambda}_{\mathcal{N}}$, we have by using Lemma 3.2, (4.31), (5.13c), and (A.1) that

$$\widehat{\Delta\lambda}_i = -\lambda_i + \mu\delta_{\mathbf{u}} + O(\mu^2), \qquad \text{for all } i \in \mathcal{N}.$$

Therefore, if $\lambda_i + \alpha\widehat{\Delta\lambda}_i = 0$ for some $i \in \mathcal{N}$ and some $\alpha \in [0,1]$, we have by arguing as in (A.4) that

$$1 - \alpha = \delta_{\mathbf{u}} + O(\mu). \tag{A.6}$$

Finally, for $i \in \mathcal{B}$, we have from Lemma 3.2 that $\lambda_i = \Theta(1)$, while from (4.31), (5.13a), and (5.13b), we have that

$$\Delta\lambda_i = O(\mu), \qquad \widehat{\Delta\lambda}_i = O(\mu) + \delta_{\mathbf{u}}/\mu, \qquad \text{for all } i \in \mathcal{B}. \tag{A.7}$$

Therefore, we have for all $\mu \gg \mathbf{u}$ that

$$\lambda_i + \alpha\widehat{\Delta\lambda}_i > 0, \qquad \text{for all } i \in \mathcal{B} \text{ and all } \alpha \in [0,1]. \tag{A.8}$$

By combining the observations (A.4), (A.5), (A.6), and (A.8), we conclude that there is a value $\hat{\alpha}_{\max}$ satisfying

$$\hat{\alpha}_{\max} \in [0,1], \qquad 1 - \hat{\alpha}_{\max} = \delta_{\mathbf{u}}/\mu + O(\mu)$$

such that

$$(\lambda, s) + \alpha(\widehat{\Delta\lambda}, \widehat{\Delta s}) > 0, \qquad \text{for all } \alpha \in [0, \hat{\alpha}_{\max}],$$

proving the claim (5.15). By making various simplifications to the analysis above, it is easy to show that (5.17) holds as well.

We now prove the claims (5.16) concerning the changes in the feasibility and duality measures along the computed step.

From (1.2), (3.11a), and the first block row of (3.8), we have

$$\begin{aligned} &r_f(z + \alpha\widehat{\Delta z}, \lambda + \alpha\widehat{\Delta\lambda}) \\ &= \mathcal{L}_z(z + \alpha\widehat{\Delta z}, \lambda + \alpha\widehat{\Delta\lambda}) \\ &= \mathcal{L}_z(z,\lambda) + \alpha\mathcal{L}_{zz}(z,\lambda)\widehat{\Delta z} + \alpha\nabla g(z)\widehat{\Delta\lambda} + O(\alpha^2\|\widehat{\Delta z}\|^2) \\ &= (1-\alpha)\mathcal{L}_z(z,\lambda) + \alpha\mathcal{L}_{zz}(z,\lambda)(\widehat{\Delta z} - \Delta z) + \alpha\nabla g_{\mathcal{B}}(z)(\widehat{\Delta\lambda}_{\mathcal{B}} - \widehat{\Delta\lambda}_{\mathcal{B}}) \\ &\quad + \alpha\nabla g_{\mathcal{N}}(z)(\widehat{\Delta\lambda}_{\mathcal{N}} - \Delta\lambda_{\mathcal{N}}) + O(\alpha^2\|\widehat{\Delta z}\|^2). \end{aligned} \tag{A.9}$$

From (4.31) and (5.13a), we have $\widehat{\Delta z} = \delta_{\mathbf{u}} + O(\mu)$, so for $\mu \gg \mathbf{u}$ and $\alpha \in [0,1]$, we have

$$\alpha^2\|\widehat{\Delta z}\|^2 = O(\mu^2). \tag{A.10}$$

From the definition (2.13) of the SVD of $\nabla g_{\mathcal{B}}(z^*)$, Theorem 3.3, and (5.13a), we have that

$$\begin{aligned}\nabla g_{\mathcal{B}}(z)(\widehat{\Delta\lambda}_{\mathcal{B}} - \Delta\lambda_{\mathcal{B}}) &= \nabla g_{\mathcal{B}}(z^*)(\widehat{\Delta\lambda}_{\mathcal{B}} - \Delta\lambda_{\mathcal{B}}) + O(\|z - z^*\|\|\widehat{\Delta\lambda}_{\mathcal{B}} - \Delta\lambda_{\mathcal{B}}\|) \\ &= \hat{U}\Sigma U^T(\widehat{\Delta\lambda}_{\mathcal{B}} - \Delta\lambda_{\mathcal{B}}) + O(\mu)\delta_{\mathbf{u}}/\mu \\ &= \delta_{\mathbf{u}}. \qquad\qquad \text{(A.11)}\end{aligned}$$

Note that the larger error (5.13b) in the component $V^T(\widehat{\Delta\lambda}_{\mathcal{B}} - \Delta\lambda_{\mathcal{B}})$, which is present when MFCQ is satisfied but not when LICQ is satisfied, does not enter into the estimate (A.11). By substituting this estimate into (A.9) together with estimates for $\widehat{\Delta z} - \Delta z$ and $\widehat{\Delta\lambda}_{\mathcal{N}} - \Delta\lambda_{\mathcal{N}}$ from (5.13), we obtain that

$$r_f(z + \alpha\widehat{\Delta z}, \lambda + \alpha\widehat{\Delta\lambda}) = (1 - \alpha)r_f + \delta_{\mathbf{u}} + O(\mu^2),$$

verifying our claim (5.16b). The potentially large error (5.13b) does not affect rapid decrease of the r_f component along the computed search direction.

For the second feasibility measure r_g, we have from (3.11b), the second block row of (3.8), and the estimates (5.13a) and (A.10) that

$$\begin{aligned}&r_g(z + \alpha\widehat{\Delta z}, s + \alpha\widehat{\Delta s}) \\ &= g(z + \alpha\widehat{\Delta z}) + s + \alpha\widehat{\Delta s} \\ &= g(z) + \alpha\nabla g(z)^T\widehat{\Delta z} + s + \alpha\widehat{\Delta s} + O(\alpha^2\|\widehat{\Delta z}\|^2) \\ &= (1 - \alpha)(g(z) + s) + \alpha\nabla g(z)^T(\widehat{\Delta z} - \Delta z) + \alpha(\widehat{\Delta s} - \Delta s) + O(\mu^2) \\ &= (1 - \alpha)r_g + \delta_{\mathbf{u}} + O(\mu^2),\end{aligned}$$

verifying (5.16c).

To examine the change in μ, we look at the change in each pairwise product $\lambda_i s_i$, $i = 1, 2, \ldots, m$. We have

$$\begin{aligned}&(\lambda_i + \alpha\widehat{\Delta\lambda}_i)(s_i + \alpha\widehat{\Delta s}_i) \\ &= \lambda_i s_i + \alpha(s_i\widehat{\Delta\lambda}_i + \lambda_i\widehat{\Delta s}_i) + \alpha^2\widehat{\Delta s}_i\widehat{\Delta\lambda}_i \\ &= \lambda_i s_i + \alpha(s_i\Delta\lambda_i + \lambda_i\Delta s_i) + \alpha s_i(\widehat{\Delta\lambda}_i - \Delta\lambda_i) + \alpha\lambda_i(\widehat{\Delta s}_i - \Delta s_i) \qquad \text{(A.12)} \\ &\quad + \alpha^2\widehat{\Delta\lambda}_i\widehat{\Delta s}_i.\end{aligned}$$

From the last block row in (3.8), the estimate $t = O(\mu^2)$ (3.7), and the estimate (4.31) of the exact step, we have

$$\lambda_i s_i + \alpha(s_i\Delta\lambda_i + \lambda_i\Delta s_i) = (1 - \alpha)\lambda_i s_i + O(\mu^2). \qquad \text{(A.13)}$$

From (4.31) and (5.13), we have

$$\widehat{\Delta\lambda}_i\widehat{\Delta s}_i = (\delta_{\mathbf{u}}/\mu + O(\mu))(O(\mu) + \delta_{\mathbf{u}}) = \delta_{\mathbf{u}} + O(\mu^2), \qquad \text{(A.14)}$$

since $\mu \gg \mathbf{u}$. For $i \in \mathcal{B}$, we have from Lemma 3.2, (5.13a), and (5.13b) that

$$s_i(\widehat{\Delta\lambda}_i - \Delta\lambda_i) = O(\mu)\delta_{\mathbf{u}}/\mu = \delta_{\mathbf{u}}, \quad \text{for all } i \in \mathcal{B}. \qquad \text{(A.15)}$$

For $i \in \mathcal{N}$, we have from Lemma 3.2 and (5.13c) that

$$s_i(\widehat{\Delta\lambda}_i - \Delta\lambda_i) = \mu\delta_{\mathbf{u}}, \quad \text{for all } i \in \mathcal{N}. \qquad \text{(A.16)}$$

For the remaining term $\lambda_i(\widehat{\Delta s}_i - \Delta s_i)$, we have from Lemma 3.2 and (5.13a) that

$$\lambda_i(\widehat{\Delta s}_i - \Delta s_i) = \delta_{\mathbf{u}}, \quad \text{for all } i = 1, 2, \ldots, m. \tag{A.17}$$

By substituting (A.13)–(A.17) into (A.12), we obtain

$$(\lambda_i + \alpha\widehat{\Delta\lambda}_i)(s_i + \alpha\widehat{\Delta s}_i) = (1-\alpha)\lambda_i s_i + \delta_{\mathbf{u}} + O(\mu^2), \quad \text{all } i = 1, 2, \ldots, m. \tag{A.18}$$

Therefore, by summing over i and using (3.5), we obtain (5.16a).

REFERENCES

[1] E. D. Andersen, J. Gondzio, C. Mészáros, and X. Xu, *Implementation of interior-point methods for large scale linear programming*, in Interior Point Methods in Mathematical Programming, T. Terlaky, ed., Kluwer Academic Publishers, 1996, ch. 6, pp. 189–252.

[2] J. Bunch and L. Kaufman, *Some stable methods for calculating inertia and solving symmetric linear systems*, Mathematics of Computation, 31 (1977), pp. 163–179.

[3] J. R. Bunch and B. N. Parlett, *Direct methods for solving symmetric indefinite systems of linear equations*, SIAM Journal on Numerical Analysis, 8 (1971), pp. 639–655.

[4] R. H. Byrd, G. Liu, and J. Nocedal, *On the local behavior of an interior-point method for nonlinear programming*, OTC Technical Report 98/02, Optimization Technology Center, January 1998.

[5] J. Czyzyk, S. Mehrotra, M. Wagner, and S. J. Wright, *PCx: An interior-point code for linear programming*, Optimization Methods and Software, 11/12 (1999), pp. 397–430.

[6] G. Debreu, *Definite and semidefinite quadratic forms*, Econometrica, 20 (1952), pp. 295–300.

[7] I. S. Duff, N. I. M. Gould, J. K. Reid, J. A. Scott, and K. Turner, *The factorization of sparse symmetric indefinite matrices*, IMA Journal of Numerical Analysis, 11 (1991), pp. 181–204.

[8] A. El-Bakry, R. A. Tapia, and Y. Zhang, *On convergence rate of newton interior-point algorithms in the absence of strict complementarity*, Computational Optimization and Applications, 6 (1996), pp. 157–167.

[9] A. Forsgren, P. Gill, and J. Shinnerl, *Stability of symmetric ill-conditioned systems arising in interior methods for constrained optimization*, SIAM Journal on Matrix Analysis and Applications, 17 (1996), pp. 187–211.

[10] R. Fourer and S. Mehrotra, *Solving symmetric indefinite systems in an interior-point method for linear programming*, Mathematical Programming, 62 (1993), pp. 15–39.

[11] J. Gauvin, *A necessary and sufficient regularity condition to have bounded multipliers in nonconvex programming*, Mathematical Programming, 12 (1977), pp. 136–138.

[12] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, third ed., 1996.

[13] N. I. M. Gould, *On the accurate determination of search directions for simple differentiable penalty functions*, IMA Journal of Numerical Analysis, 6 (1986), pp. 357–372.

[14] N. I. M. Gould, D. Orban, A. Sartanaer, and P. Toint, *Superlinear convergence of primal-dual interior-point algorithms for nonlinear programming*, Technical Report TR/PA/00/20, CERFACS, April 2000.

[15] W. W. Hager, *Convergence of Wright's stabilized SQP algorithm*, Technical Report, Department of Mathematics, University of Florida, Gainesville, Fl., January 1997.

[16] ———, *Stability in the presence of degeneracy and error estimation*, Technical Report, Department of Mathematics, University of Florida, Gainesville, Fl., January 1997.

[17] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, SIAM Publications, Philadelphia, 1996.

[18] ———, *Stability of the diagonal pivoting method with partial pivoting*, SIAM Journal on Matrix Analysis and Applications, 18 (1997), pp. 52–65.

[19] O. L. Mangasarian and S. Fromovitz, *The Fritz-John necessary optimality conditions in the presence of equality and inequality constraints*, Journal of Mathematical Analysis and Applications, 17 (1967), pp. 37–47.

[20] S. Mehrotra, *On the implementation of a primal-dual interior point method*, SIAM Journal on Optimization, 2 (1992), pp. 575–601.

[21] R. D. C. Monteiro and S. J. Wright, *Local convergence of interior-point algorithms for degenerate monotone LCP*, Computational Optimization and Applications, 3 (1994), pp. 131–155.

[22] D. Ralph and S. J. Wright, *Superlinear convergence of an interior-point method despite dependent constraints*, Preprint ANL.MCS-P622-1196, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Ill., November 1996. To appear in Mathematics of Operations Research.

[23] D. Ralph and S. J. Wright, *Superlinear convergence of an interior-point method for monotone variational inequalities*, in Complementarity and Variational Problems: State of the Art, M. C. Ferris and J. Pang, eds., SIAM Publications, Philadelphia, Penn., 1997, pp. 345–385.

[24] M. H. Wright, *Ill-conditioning and computational error in interior methods for nonlinear programming*, SIAM Journal on Optimization, 9 (1998), pp. 84–111.

[25] S. J. Wright, *Stability of linear equations solvers in interior-point methods*, SIAM Journal on Matrix Analysis and Applications, 16 (1994), pp. 1287–1307.

[26] ———, *Modifying SQP for degenerate problems*, Preprint ANL/MCS-P699-1097, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Ill., October 1997.

[27] ———, *Primal-Dual Interior-Point Methods*, SIAM Publications, Philadelphia, 1997.

[28] ———, *Stability of augmented system factorizations in interior-point methods*, SIAM Journal on Matrix Analysis and Applications, 18 (1997), pp. 191–222.

[29] ———, *Effects of finite-precision arithmetic on interior-point methods for nonlinear programming*, Preprint ANL/MCS-P705-0198, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Ill., January 1998.

[30] ———, *Superlinear convergence of a stabilized SQP method to a degenerate solution*, Computational Optimization and Applications, 11 (1998), pp. 253–275.

[31] ———, *Modified Cholesky factorizations in interior-point algorithms for linear programming*, SIAM Journal on Optimization, 9 (1999), pp. 1159–1191.

[32] S. J. Wright and D. Ralph, *A superlinear infeasible-interior-point algorithm for monotone nonlinear complementarity problems*, Mathematics of Operations Research, 21 (1996), pp. 815–838.